

大規模計測点群のための 形状処理技術

東京大学大学院工学系研究科
システム創成学専攻
増田 宏

1

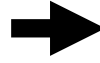
本日の内容

1. イントロダクション
2. 点群の描画
3. ノイズ処理
4. メッシュ生成
5. 曲面計算
6. フィーチャモデリング

2

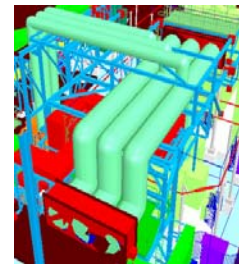
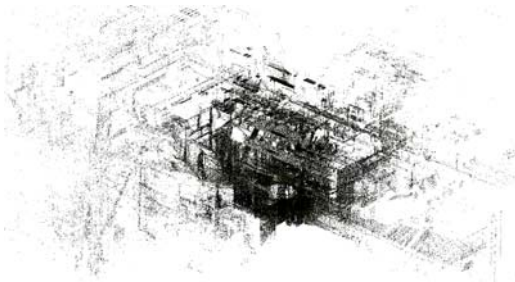
レーザ計測を用いた3Dデータの取得

- 人工物の計測
 - 工場, プラント
 - 大型機械, 船舶,
 - 建築, 土木, 都市,
 - 道路, 橋,



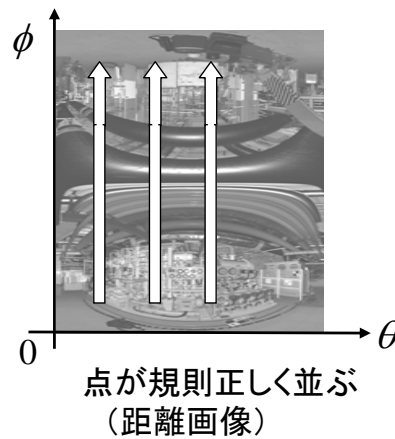
- 3Dモデル
 - 点群モデル
 - メッシュモデル
 - ソリッドモデル

⇒ シミュレーション



3

中長距離のレーザ計測装置



隣り合った点をつなげば
メッシュモデル

レーザ計測による
点群の取得

規則正しく
並んだ点群

処理が比較的
簡単

4

2. 点群の描画

● 3次元形状の表現

- 曲面の集合
- 三角形の集合
- 点の集合

要素数： 多くなる
アルゴリズム： 単純になる

CPU性能 & GPU性能
メモリ大容量化

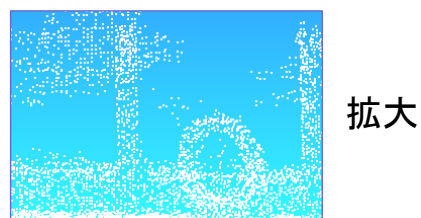
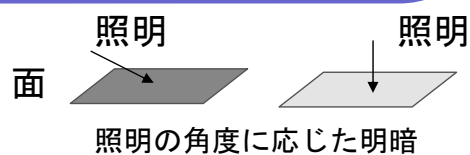
● 画面への表示

- ピクセルレベルの解像度があれば十分.

5

点の描画（色なし点群）

- 面：法線を持つ
- 点：座標のみ：
明暗が変化しない.



点だけでは、対象物が
わからない.

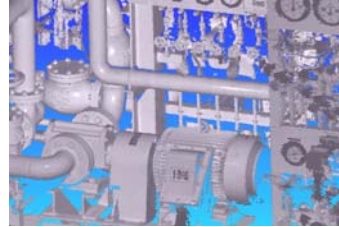
6

点の描画（色付き点群）

● 色付き点群

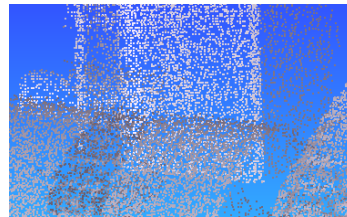
- 点データ：座標＋色
- レーザ計測
 - 明暗値：反射強度
 - 色：デジタル画像との合成

レーザー光



● 問題点

- 計測時の反射強度を明るさを変えられない。
- 描画処理では、ライトの設定や視点が反映されない。

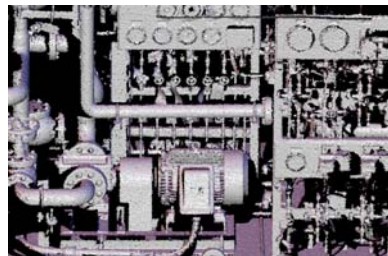


7

点の描画（法線付き点群）

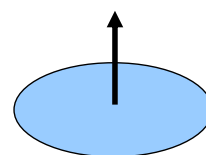
● 法線付き点群

- 点データ：座標＋法線
 - 法線は近傍点を使って計算する。
- ライトの設定や視点に応じて、描画の明暗が変化する。

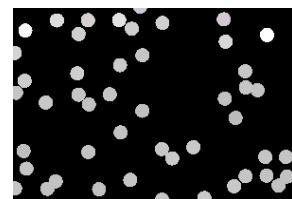


● 問題点

- 点の粗密への対応
- 拡大・縮小への対応
 - 近付くと点の集まり。



点⇒微小な面

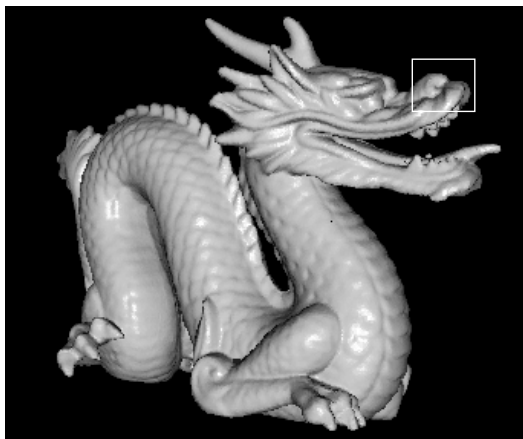


拡大図

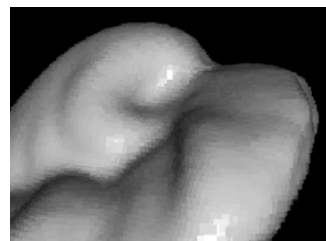
8

点の描画（法線付き点群+適応的制御）

- Qsplat（プログラムが公開）
 - <http://graphics.stanford.edu/software/qsplat/>
 - 距離に応じて、描画点数や点の大きさを変更する。



制御なし

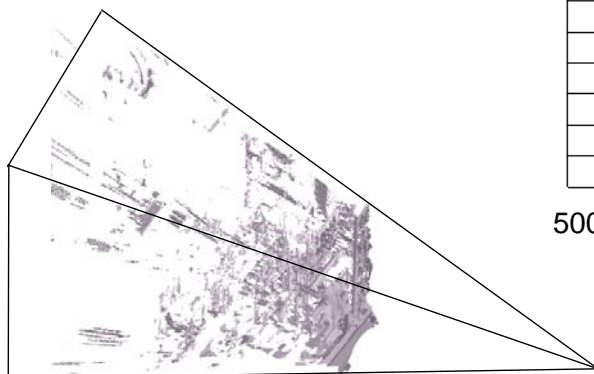


制御

9

3. 点群処理

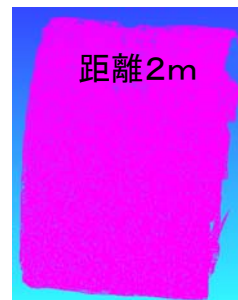
- 点群密度は距離に応じて大きく変わる



単位面積あたりの点の個数は、
距離の2乗に反比例

距離(m)	ピッチ(mm)
1	0.6
2	1.3
3	1.9
4	2.5
5	3.1
10	6.3
15	9.4
20	12.6
25	15.7

5000万点を計測した場合

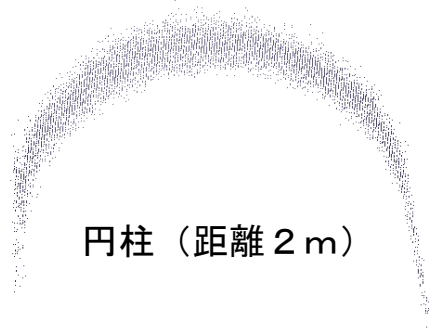


距離に応じて
- 近傍点の個数を変える
- 間引き幅を変える

10

点群処理（ノイズ処理）

- 計測点にはノイズが多い。
 - 計測点をそのまま使うと曲面計算が不安定になる。



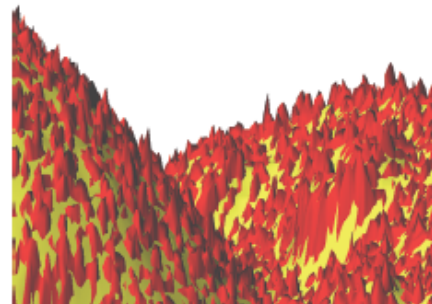
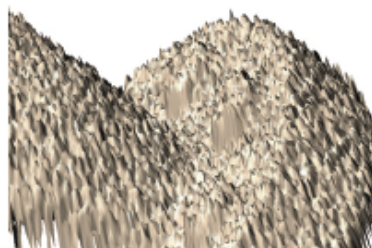
→
平滑化



11

平滑化の方法

- 高周波除去フィルタ
 - 周波数の高い信号と低い信号が混在するとき、低周波の信号のみを取り出す。
- 2次曲面当てはめ
 - 点群が滑らかな2次曲面上に乗っていると見做して、位置を補正する。



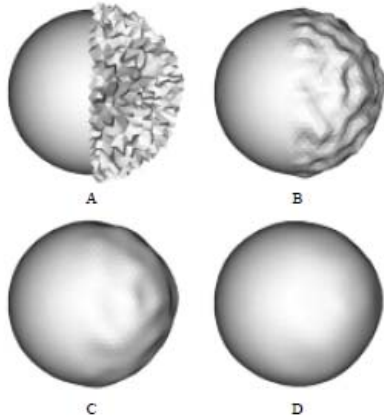
12

高周波除去フィルタ

● ラプラシアンフィルタ

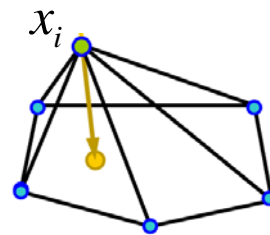
● G. Taubin (SIGGRAPH 1995)

- A Signal Processing Approach To Fair Surface Design
- 必要以上に平らにならない手法を提案



$$\Delta x_i = \sum w_j (x_j - x_i)$$

離散ラプラス演算



13

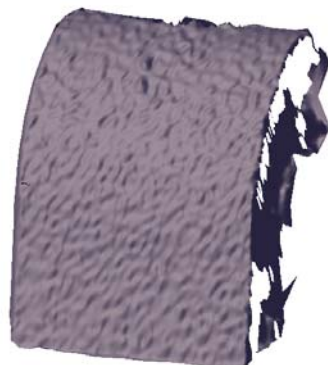
高周波除去フィルタ

● Taubin のフィルタ

- ノイズの振幅が点の間隔に比べて大きいとノイズが取れない！



計測点群



フィルタ × 50回
(これ以上改善しない)



点群を間引いてから
フィルタ × 50回

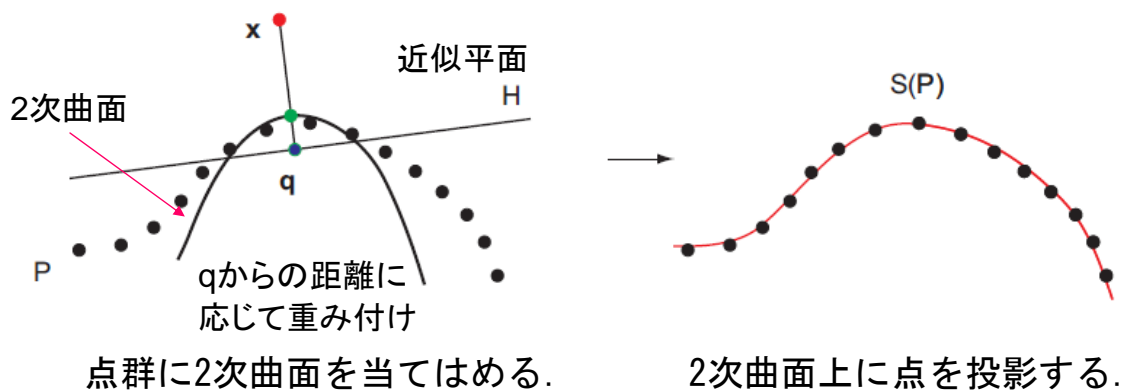
14

高周波除去フィルタを用いた平滑化



15

曲面近似による平滑化



点の粗密があるので、近傍の点の個数は、距離に応じて変える。

最小2乗法は、異常値に弱い ⇒ ロバスト推定

16

ノイズや異常値の多いデータの平滑化

- ロバスト推定（最尤推定）
 - 計測値の誤差から，元はどんな曲面上に乗っていた可能性が高いかを推定する。
 1. 曲面式 $S(x)=0$ が決まれば，計測値 x_i の誤差 r_i が決まる.
 2. 誤差 r_i は，ある確率分布 f に従うとする.
 3. 確率分布が決まれば，計測値の誤差 $\{r_i\}$ が発生する確率が計算できる. 確率を最大にする曲面式は，以下の式で求まる.

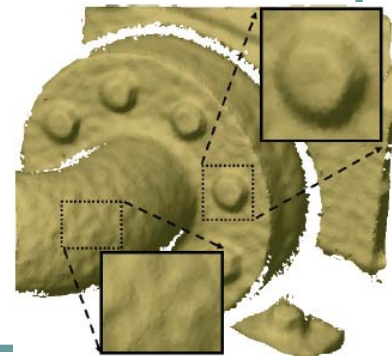
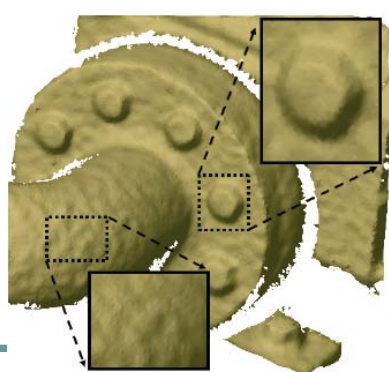
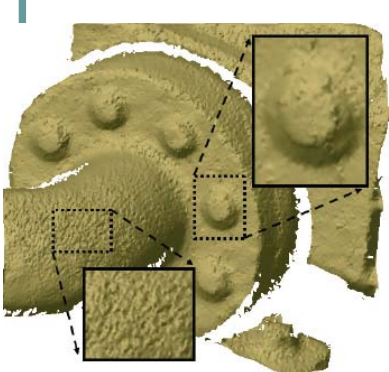
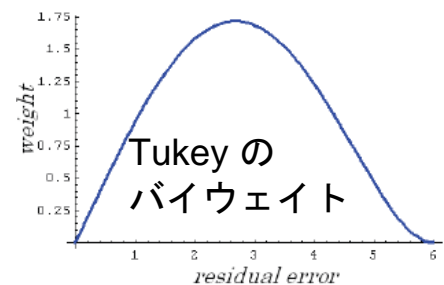
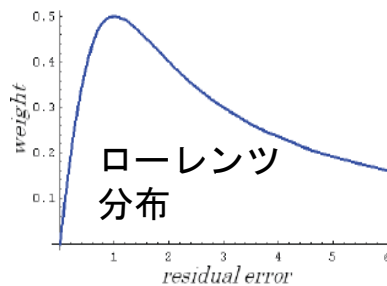
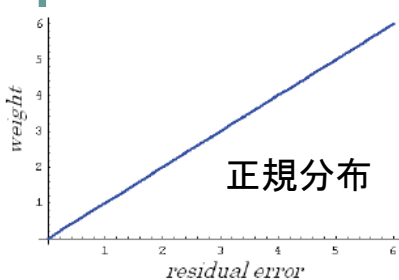
$$\arg \max_a \left[\prod_{i=1}^N f(r_i) \right]$$

⇒ 誤差分布を仮定することで曲面が求まる.

17

確率分布と計算される曲面式

大きな誤差が結果にどれだけ影響を与えるか？



19

平滑化の計算時間

● 計算時間の比較

- 単位時間あたりに処理できる点の個数

		points/sec	
	MLS	2143	1.0
ノイズ に強い	Lorentzian estimation	1150	0.5
	Tukey's biweight	1654	0.8

20

4. メッシュ生成

● 点群からのメッシュ生成

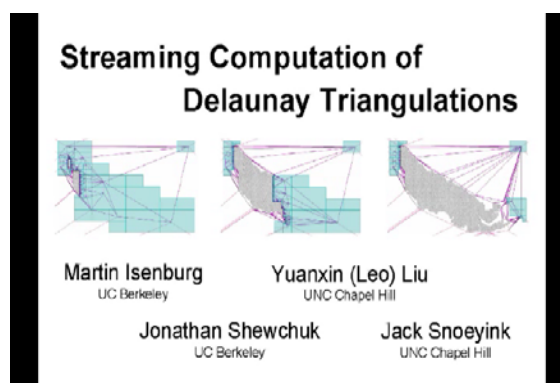
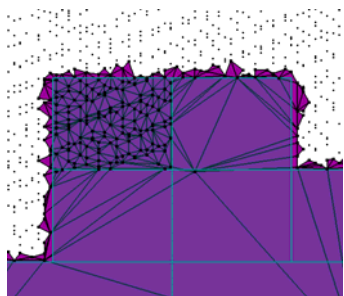
1. 計測点の並びを利用する.
 - 処理は最も簡単で高速.
 - 粗密の補正などで隣接関係が崩れると計算できない.
 - 単一の計測データのみで有効.
2. ドロネー三角形分割を行う.
 - 計算法がよく知られている.
 - 点の欠落や選択的な間引きに対応できる.
 - 平面への写像が必要.
3. 陰関数を用いる.
 - 複数データをマージしても処理できる.
 - 解像度を上げると計算コストが大幅に増大する.
 - 大きな欠落のない閉じた形状しか対応できない.

21

ドロネー三角形分割

- 大規模点群の三角形分割（ソースが公開）

- M. Isenburg (2006)
 - Streaming Computation of Delaunay Triangulations
- <http://www.cs.unc.edu/~isenburg/sd/>



22

陰関数を利用したメッシュ生成

- 陰関数

$$f(x, y, z) = 0$$

で立体の表面を表現する。

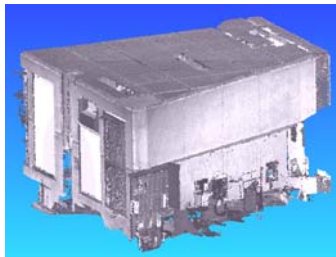
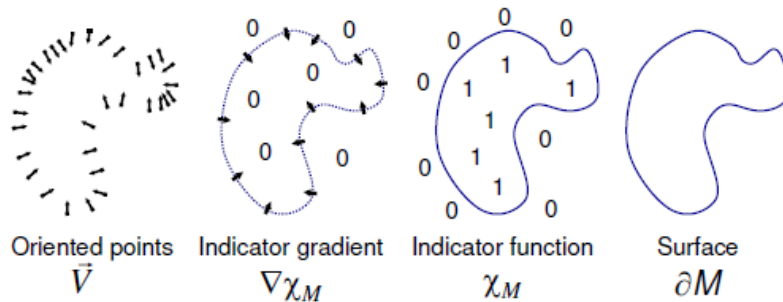
- 点群の座標や法線が与えられたときに、それらを満たす陰関数曲面を計算する。
 - 近年では高速な計算手法が提案されている。
 - Y. Ohtake: Sparse Low-degree Implicit Surfaces with Applications to High Quality Rendering, Feature Extraction, and Smoothing
 - www.riken.go.jp/lab-www/V-CAD/members/ohtake/slim/slim.pdf
 - M. Kazhdan: Poisson Surface Reconstruction
 - <http://www.cs.jhu.edu/~misha/Code/PoissonRecon/>

23

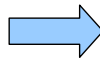
陰関数を利用したメッシュ生成

- Poisson Surface Reconstruction (ソースが公開)

入力は
法線付き
点群



増田研究室の点群

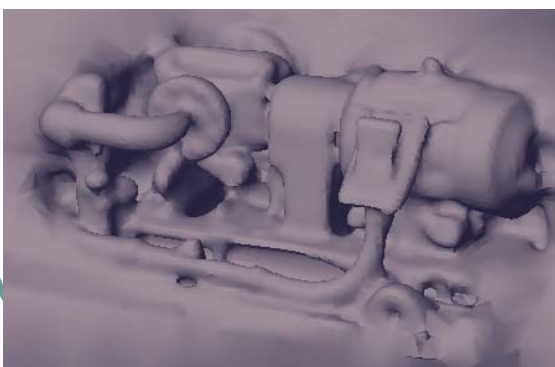
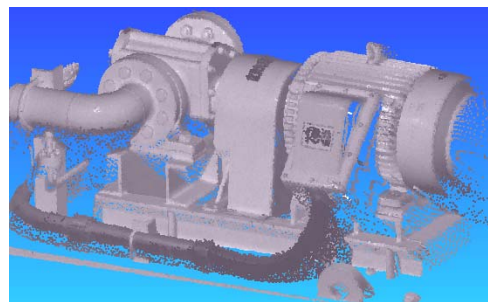


研究室のメッシュモデル

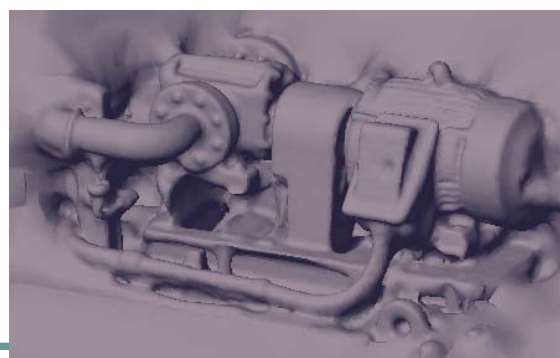
24

陰関数を利用したメッシュ生成

- 利点
 - 穴を自動的に塞ぐ
- 欠点
 - 解像度に限界
 - 粗密や欠落のある点群は苦手
 - 形状に予期できない不具合.



低解像度

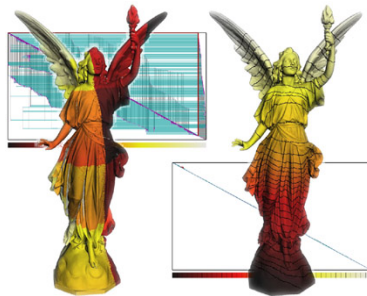


高解像度

25

大規模メッシュの表現法 Streaming Mesh

- 大規模メッシュをファイルの最初から逐次処理するためのフォーマット
 - Isenburg, 2005
 - 頂点、面情報の混在
 - 利用されなくなった頂点のメモリを開放

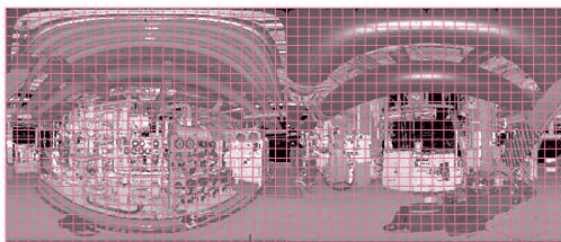


```
model.sma
v(1.0, 0.0, 0.0)
v(1.0, 2.0, 1.0)
v(1.5, 2.0, 1.4)
f(1, 2, 3)
v(1.5, 2.0, 1.4)
f(1, 2, 4)
f(-1, 3, 4)
v(2.0, 0.4, 3.0)
f(2, -3, 5)
⋮
```

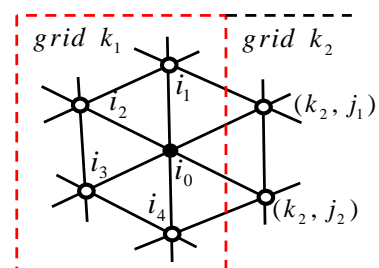
26

大規模メッシュの表現法 Grid Mesh

- 大規模メッシュのランダムアクセスが可能な表現
 - 当研究室で開発
 - 頂点をグリッドに分割する.
 - 頂点を (グリッドID, 頂点ID) でラベル付けする.
 - 同じグリッドの頂点は, HDの連続領域に記録する.



グリッド

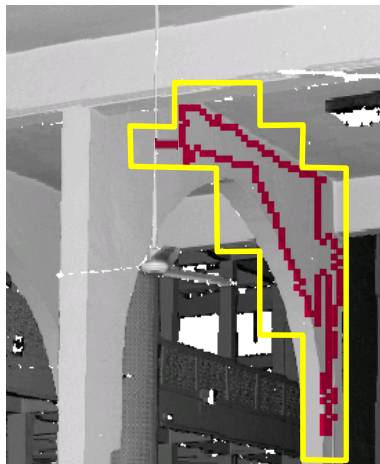


隣接関係の表現

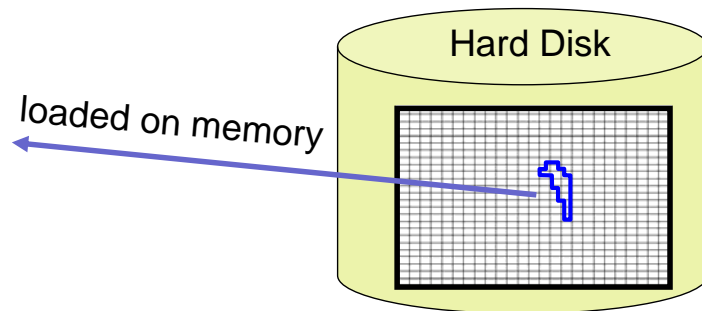
27

Grid Mesh

- 必要なデータだけをメモリ上に読み込む。



平面領域の探索



28

5. 曲面計算

- 部材の寸法（直径など）を得たいとき
 - 点群 $\{p_i\}$ を近似する多項式曲面を計算する。
 - 平面，円柱，円錐，球，トーラスなど
- 2次形式で表現された曲面
 - 計算は単純だが，パラメータが多すぎる。
⇒ 計算が不安定．誤差が大きい。

$$(x \ y \ z \ 1) \begin{pmatrix} a & e & f & g \\ e & b & h & i \\ f & h & c & j \\ g & i & j & d \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = 0$$

29

曲面計算の方法

$$1. \sum_i \{f(x_i, y_i, z_i)\}^2 \rightarrow \min$$

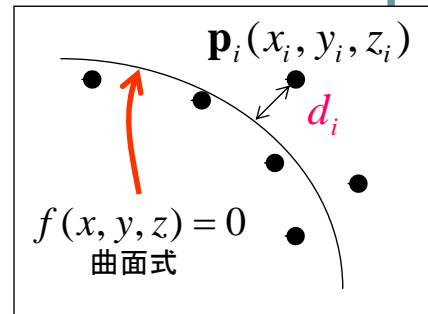
- 解がfの選び方で大きく変わる.

$$2. \sum_i d_i^2 \rightarrow \min$$

- 平方根を含み, 計算が不安定.

$$3. \sum_i \{g(\mathbf{p}_i) - h(\mathbf{p}_i)\}^2 \rightarrow \min$$

- ノイズに敏感.



解析曲面と点Pの距離

$$d_i(\mathbf{p}_i) = \sqrt{g(\mathbf{p}_i) - h(\mathbf{p}_i)}$$

平方根を持つ項と
持たない項に分けられる.

曲面計算の方法

$$4. \sum_i \tilde{d}_i^2 \rightarrow \min$$

$$\tilde{d}_i(\mathbf{p}_i) = \frac{g(\mathbf{p}_i) - h^2(\mathbf{p}_i)}{2h^2(\mathbf{p}_i)}$$

距離の2乗の
最小化問題を
よく近似する

$d(\mathbf{q})=0$ のとき

$$\tilde{d}(\mathbf{q})=0$$

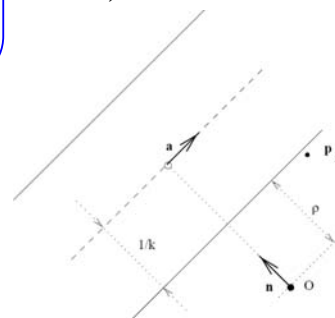
$$\frac{\partial d}{\partial q_i}(\mathbf{q}) = \frac{\partial \tilde{d}}{\partial q_i}(\mathbf{q})$$

修正距離関数

G. Lukacs, et al.: Geometric least-squares fitting of spheres, cylinders, cones and tori, RECCAD, 1997

円柱の修正距離関数

$$\tilde{d}(s, \mathbf{p}) = \frac{k}{2} |\hat{\mathbf{p}} \times \mathbf{a}|^2 - \langle \hat{\mathbf{p}}, \mathbf{n} \rangle$$



曲面の計算

- 修正距離関数を用いた非線形最適化

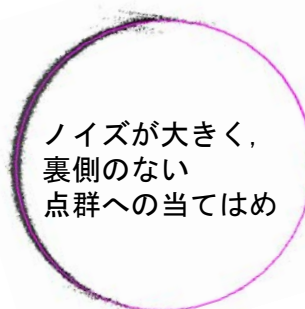
- Gauss-Newton 法
- Levenberg-Marquardt 法

- 実験結果

- ○ 球面 (1/2)
- ○ 円柱面 (1/2)
- △ 円錐面 (1/2)
- × トーラス面 (1/8)

⇒ 精度のよい計算には、
形状に対する知識や
ユーザの指示が必要

自由度				
平面	球面	円筒面	円錐面	トーラス
3	4	5	6	7



32

6. フィーチャモデリング

- フィーチャ：工学的な意味を持った形状

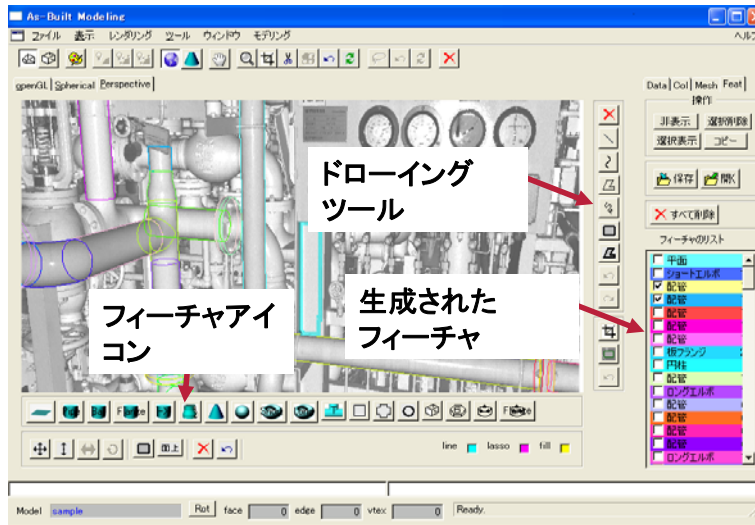
- 配管：
 - パイプ, フランジ, エルボ, レジューサ, など
- 鋼材
 - H型鋼, I型鋼, 等辺山型鋼, など
- 建築
 - 窓, ドア, 階段, など
- 道路
 - 道路, ガードレール, 標識, 信号, など

- フィーチャモデリング

- フィーチャを認識したり, 組み合わせたりして
形状モデリングを行う.

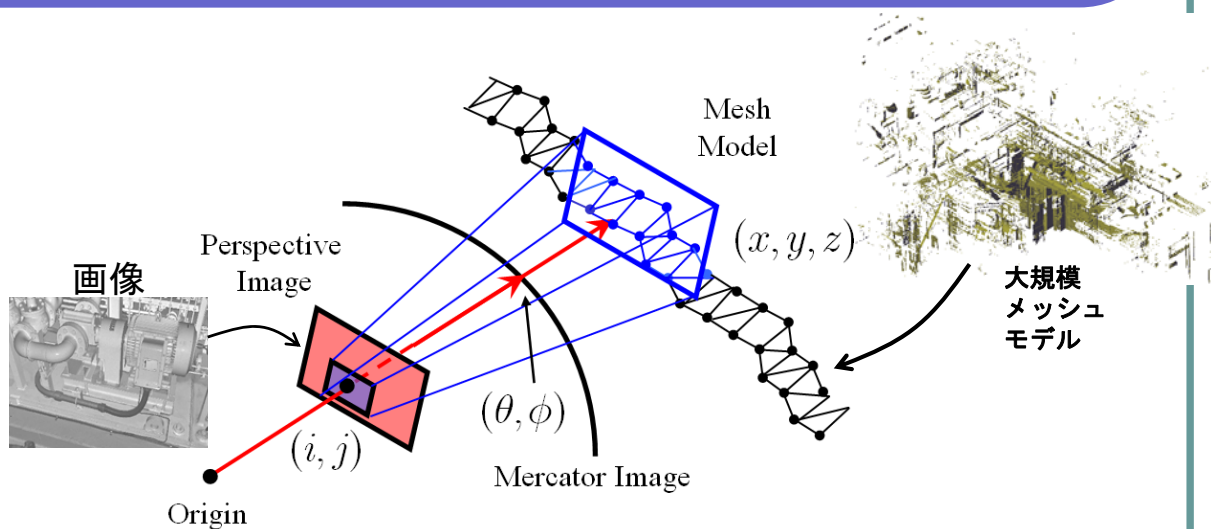
33

フィーチャベースモデリング



34

大規模メッシュへの画像インタフェース



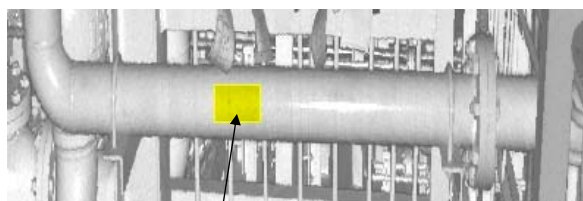
人は画像を見てモデリング
コンピュータは点群を使って幾何計算

35

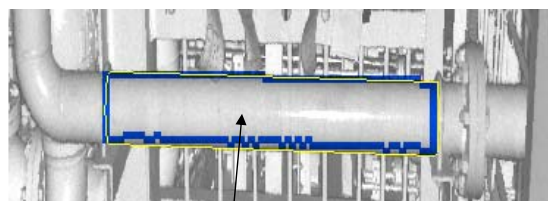
領域成長法によるフィーチャ抽出

● 手順

- シード領域をユーザが指定
- シード領域に含まれるメッシュ頂点を求める.
- メッシュ上で, 曲面上にある近傍の点を追加.
- 曲面の再計算.



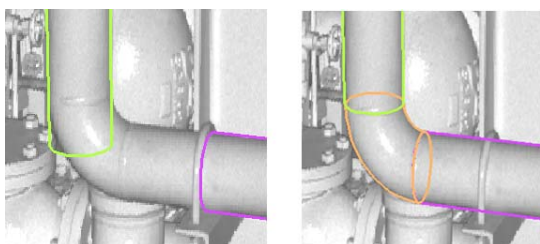
シード領域



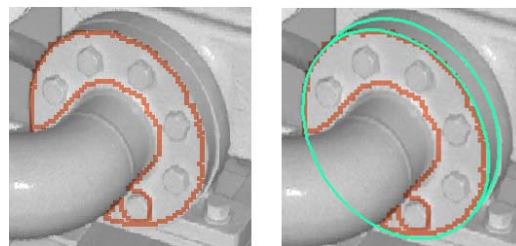
検出領域

36

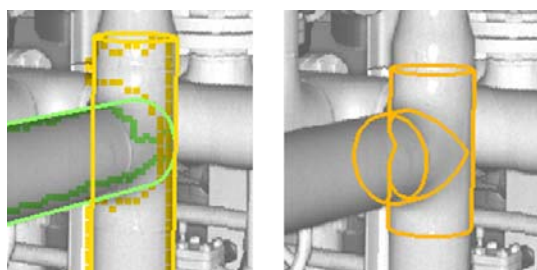
フィーチャの作成



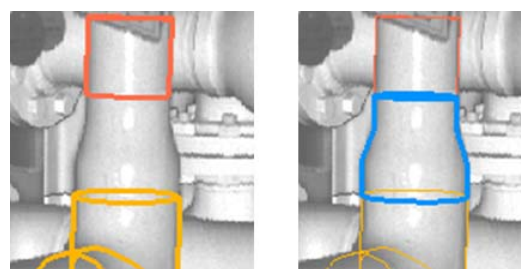
エルボ



フランジ



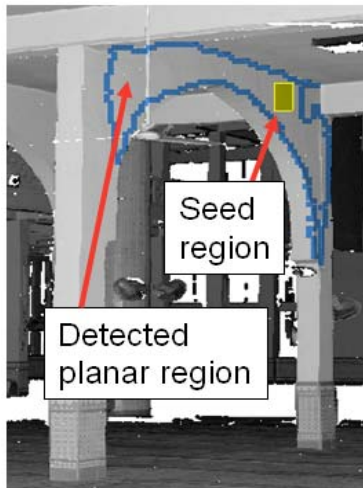
ティー



レジュース

37

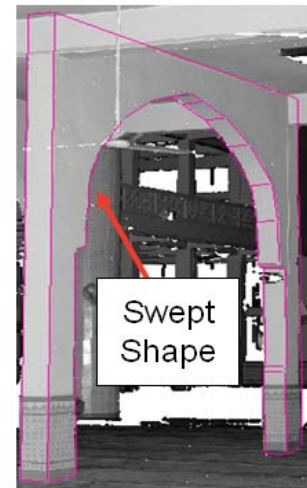
多角柱フィーチャの作成



平面の検出



スケッチ

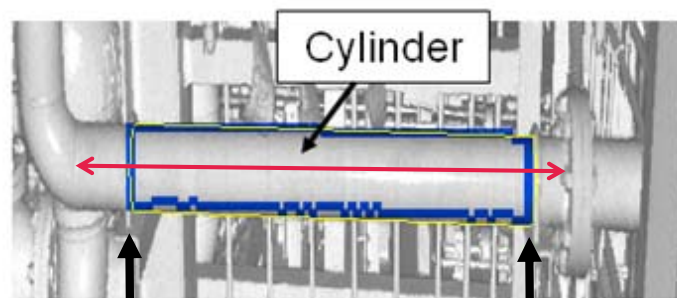


多角柱の作成

38

フィーチャの編集

- 他の部品に隠されると、フィーチャ領域が完全には抽出できない。

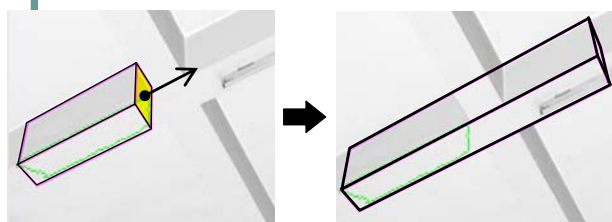


固定金具

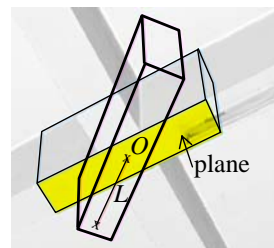
固定金具

39

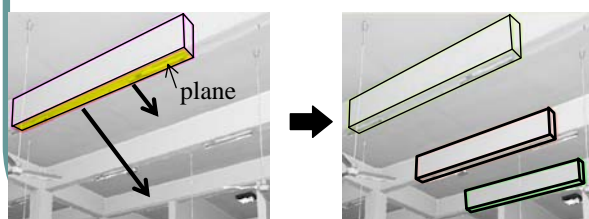
フィーチャの編集操作



寸法変更



回転



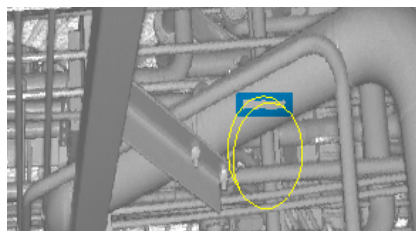
コピーと移動

インタラクティブに操作ができる。

40

曲面計算の安定性

- 計算が失敗しやすいケース
 - 他の部品に隠されている
 - ⇒ 計算が不安定
 - 遠い部材, 黒っぽい部材
 - ⇒ 少ない点群, 大きいノイズ
 - 細い部材 ⇒ 径が不正確
 - 短い部材 ⇒ 軸が不正確



41

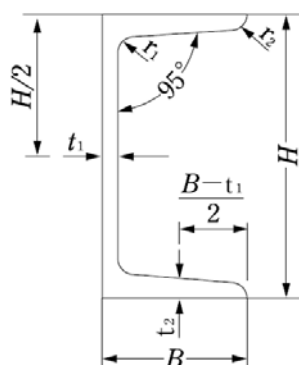
計算精度向上のための制約

- 工業規格
 - 離散的な値を取ることを利用
- ユーザスケッチ
 - 補助的な情報
- 他部品との位置関係
 - 制約の追加

42

規格寸法

- 工業規格で定まっている寸法がある
 - ANSI, JIS
 - 最も近い寸法を用いて再計算する



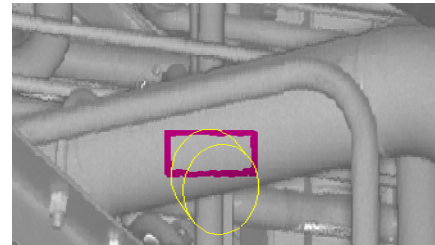
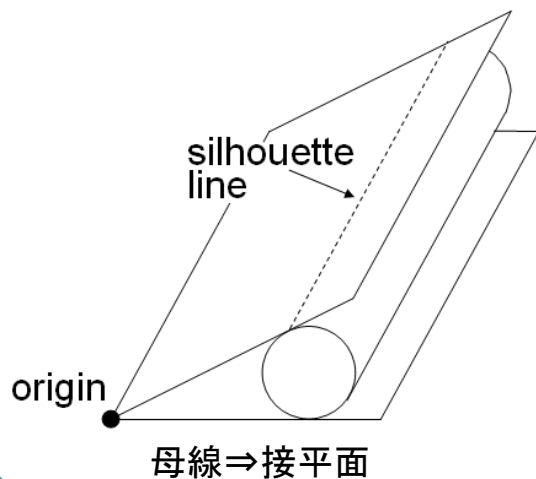
H×B	t ₁	t ₂	r ₁	r ₂
100×50	5	7.5	8	4
125×65	6	8	8	4
150×75	9	12.5	15	7.5
180×75	7	10.5	11	5.5

The JIS Standard

43

ユーザスケッチ

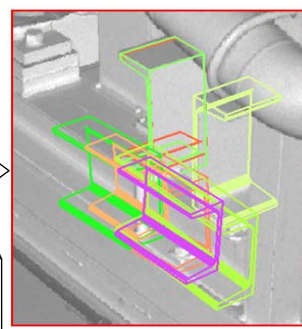
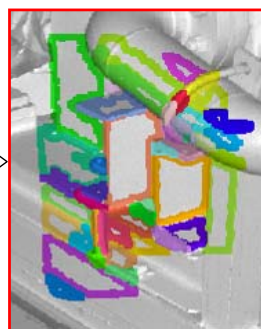
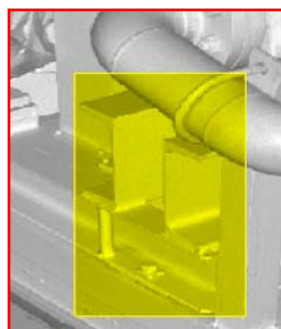
- 円筒，円錐の母線を画面上でスケッチ
 - 未知数が減る



44

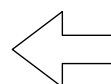
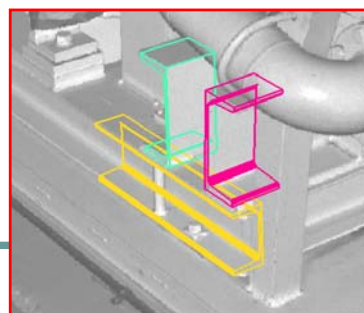
インテリジェントフィーチャ

- 指定した範囲から自動的に鋼材を探索する。



工業規格

算出された
ソリッドモデル



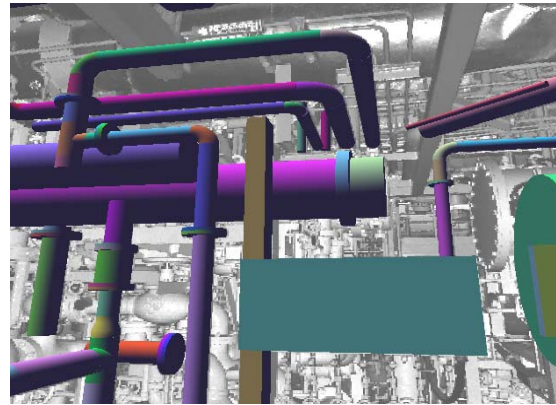
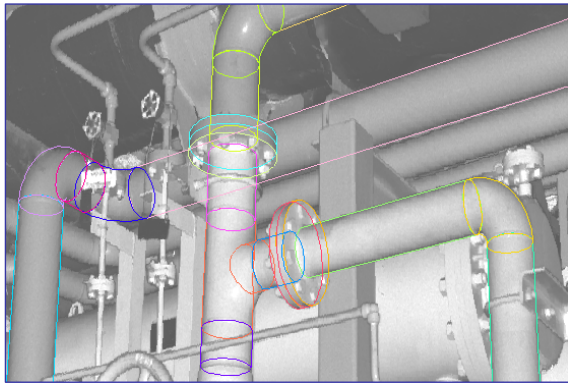
評価式



45

例題1

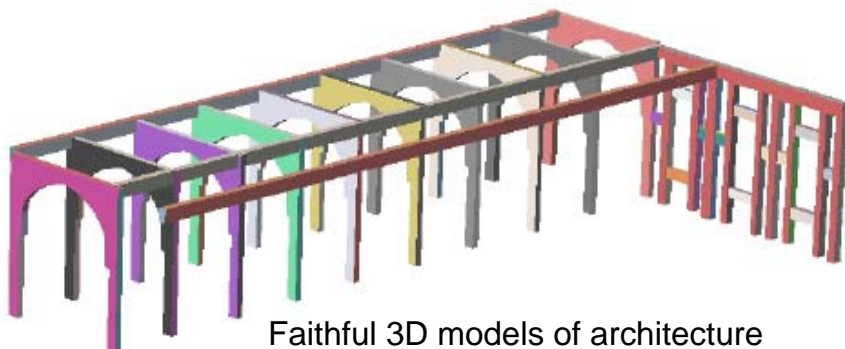
- プラント



例題2

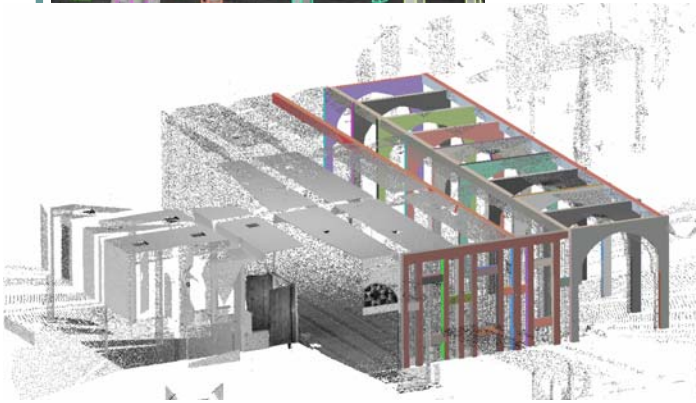
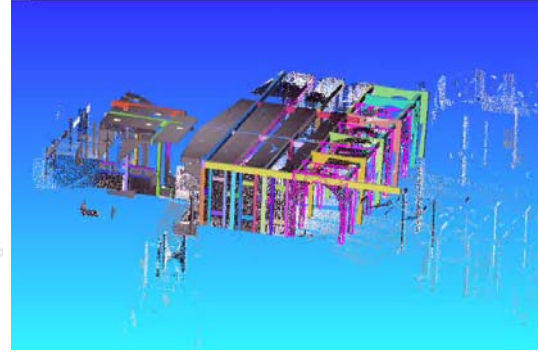
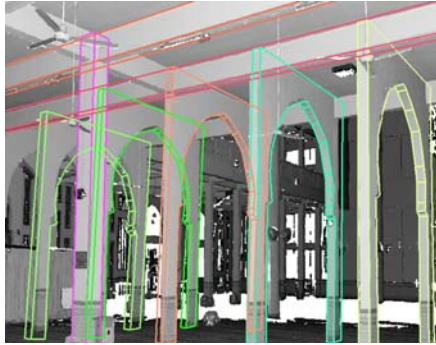


Perspective image



Faithful 3D models of architecture

例題2（その2）



48

まとめ

- 点群処理のための基本的な技術を紹介した。
- 本研究室で開発した手法や形状モデリングシステムを紹介した

49