ワークショップ
「点群処理システムの課題」

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

ワークショップ「点群処理システムの課題」

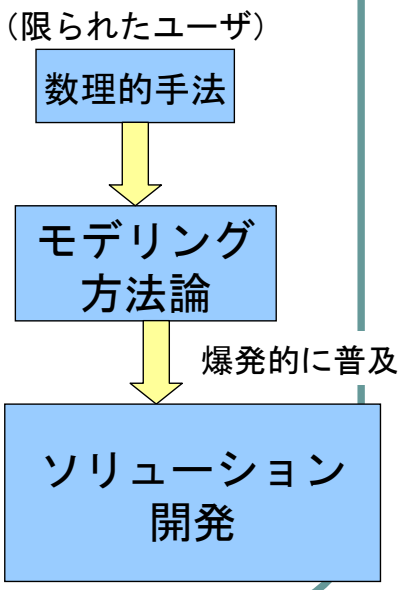
- 点群処理の基盤技術は既にある.
  - ＣＡＤ，コンピュータグラフィクス，コンピュータビジョン
    - 点群描画，平滑化，メッシュ生成，レジストレーション，セグメンテーション，補間手法，自動認識など.
- これらは，部品に過ぎない.
  - ユーザが欲しいのは「使いやすい完成品であるシステム」
- 大規模人工物の点群処理システム
  - システムとしてはまだ発展の余地が多い.
  - 「システム」としての最適な姿が見えていない.

技術の進化 ⇔ マーケットの育成

- 現状の課題
  - マーケットの拡大
  - ソリューションベンダの拡大
    ⇒　多様なニーズに応えるシステム
- ワークショップ
  - 技術課題の発信
  - 技術成果の発信
  - 情報交換の場

機械系CAD: ソリッドモデル

- 1974～1980年代
  - 基盤技術がほぼ出揃う
    - そのころのシステム：
      - 座標をユーザが打ち込む
      - 基本立体を足したり引いたりする.
- 1987～1990年代
  - 信頼性と効率の向上
  - 使い勝手を高めるための技術
    - フィーチャベースモデリング
    - パラメトリックモデリング
- 2000年以降
  - ソリューション開発
  - コンサルティング
  - システムを使いこなす技術
  - ユーザ教育



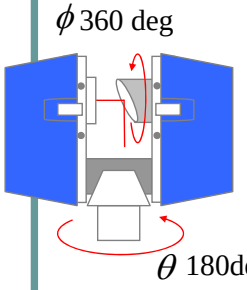
レーザ計測分野の研究開発の活性化

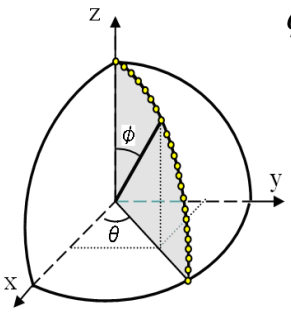
- 多くの研究者に参加してもらうには：
  - データの入出力を容易にする。
    - 標準フォーマット
  - 多様なサンプルデータを用意する。
    - CGは、サンプルデータが豊富
    - 誰でも簡単に参入できる。
  - ベンチマークデータを決める。
    - 性能の比較：精度、範囲、自動化率、工数

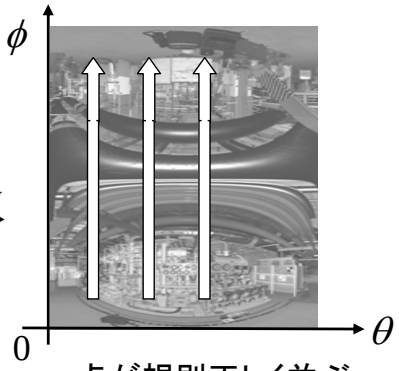
研究分野

- メッシュ生成
- レジストレーション
- モデリング方法論
- 平滑化
- 大規模点群処理
- 曲面フィッティング
- セグメンテーション
- 形状修復
- 自動検出、形状検索
- 並列処理

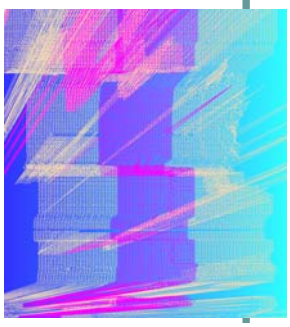
メッシュ生成







点が規則正しく並ぶ
（距離画像）



隣り合った点を
つなげば
メッシュモデル

計測した順に点を並べて画像を作成する方法は，1980年以前からある古典的手法 → 画像処理の問題に帰着 → 複雑な3D処理をしなくても問題が解ける

メッシュ生成に関連する技術

- **ストリーミング技術**
  - **大規模点群を計測順に処理してメッシュを生成する技術。地形計測のために開発。**
  - **数GBの点群をノートPCで処理可能。**
- **メッシュの併合**
  - **ミケランジェロプロジェクト**
- **メッシュ簡単化**
  - **平面は粗く間引く。遠い部品、曲率の大きい部分は細かく。**
  - **巨大データの処理が課題。**
- **陰関数処理**
  - **F (x,y,z) = 0 で表現。**
  - **Poisson Surface Reconstruction**

レジストレーション

- 方法論
  - マーカ
    - 計測の手間がかかる。
  - ICP
    - 点群が合うところを探す。古典的な手法。計算時間がかかる。
  - 特徴量
    - 曲率、平面などを検出して、位置合わせ。計算が速い。
- 計算
  - 初期値あり
    - ユーザが近くまで手動で合わせる。
  - 初期値なし
    - 処置位置を自動的に決める。
- 精度
  - 逐次型： ずれが蓄積する。
  - 同時複数型： 全体を一括して計算する。データ量が問題。
- 変形
  - 剛体として位置合わせ。
  - アフィン変換 → 誤差を分散させる。

点群のフォーマット

- xyz, pts など
  - (X, Y, Z, I, RGB) のデータ
  - 画像への変換には，計測ピッチが必要.
- ptx (Leica) ← これが便利
  - (X, Y, Z, I, RGB) のデータが計測順に並ぶ
  - 画像への変換が簡単
- ベンダ固有のバイナリデータ
  - zfs、fws など. ライセンスが必要.

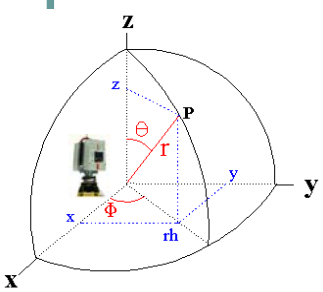
モデリング方法論1

- 通常，複数地点から計測する．
  - レジストレーション　（位置合わせ）

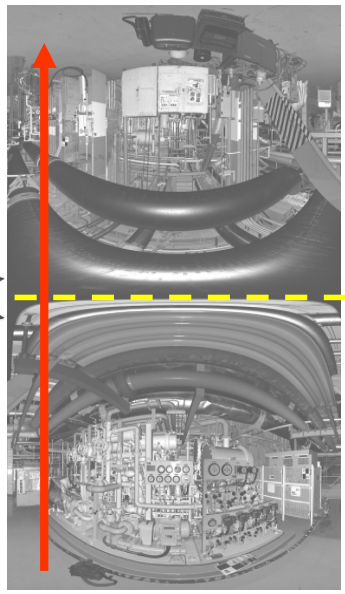
↓

- 点群モデリングの流儀
  - 複数の点群を混ぜてからモデリング　（ＣＧ）
    - 3次元処理が必要．ボクセルなど．
    - 比較的複雑な処理になる．
  - モデリングしてから複数の点群を混ぜる．（ＣＶ）
    - 画像処理問題に帰着．
    - 処理が圧倒的に簡単．

モデリング方法論2



$(\phi, \theta, r) \Leftrightarrow (x,y,z)$



生データ

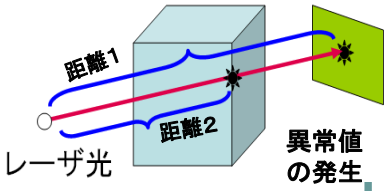
パノラマ画像

点群＋イメージベースモデリング
画像ベースの３D処理

データの平滑化

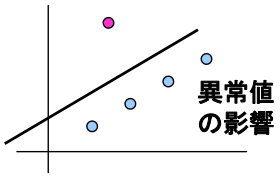
大量の異常値・大きなノイズ

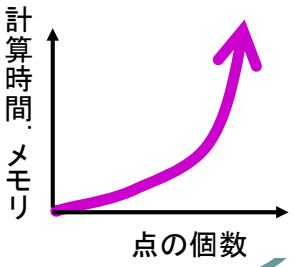
- 複数面からの反射などが原因.
- 曲面計算は，異常値に弱い.



点の個数が膨大

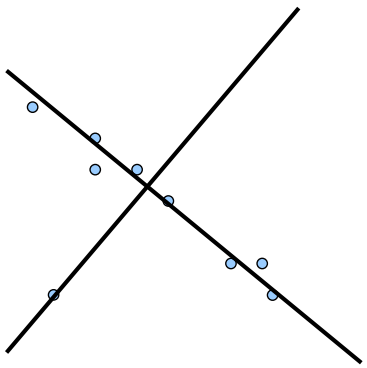
- 大域的な手法(全体最適)は利用が難しい。





最小2乗法

- ずれの2乗和を最小にする。



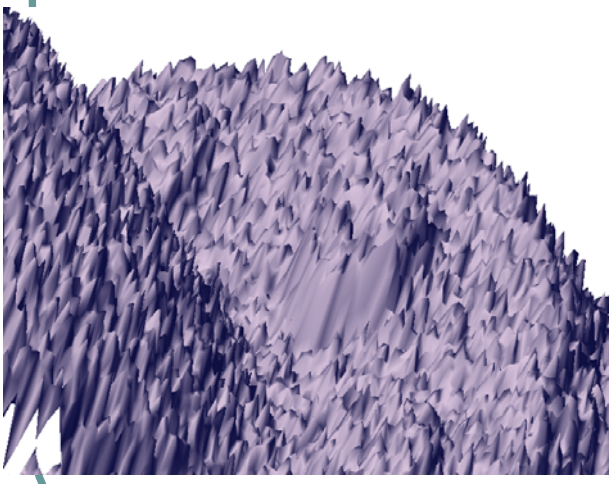
・あらかじめ異常値を除去する。
・異常値に強い手法を用いる。

データの平滑化

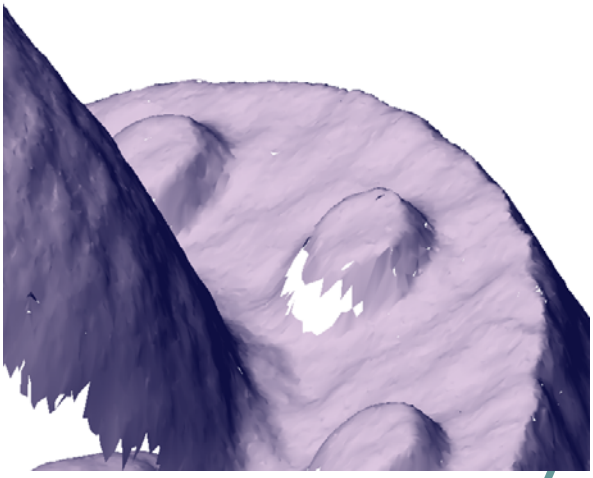
- 平滑化
  - 高周波成分除去(Gaussian フィルタなど)
    - 高速 (普通の計算機でも高速).
    - 形を多少変える.(やや平らになる)
    - 見栄えをよくするときに有効.
  - 局所曲面近似
    - 計算時間がかかる.
    - 本来の形状に忠実.
    - 寸法値を正確に得るときに有効

今回は, CADモデル化が目的なのでこちらを使う.

データの平滑化

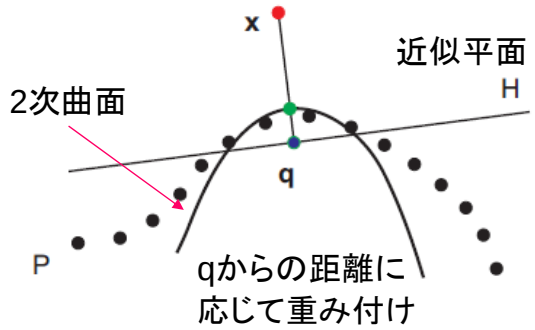


元データ

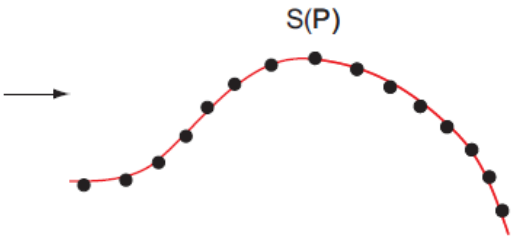


平滑化後のデータ

曲面近似による平滑化



点群に2次曲面を当てはめる.



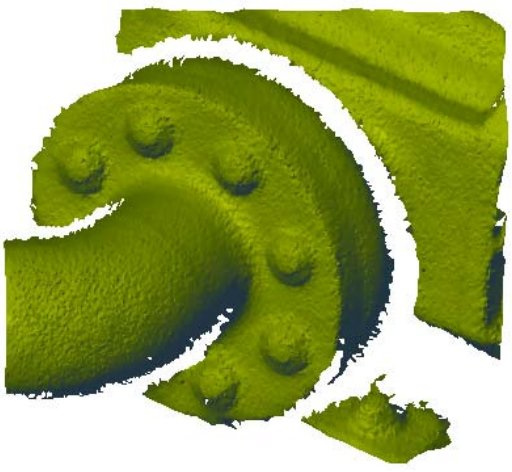
2次曲面上に点を投影する.

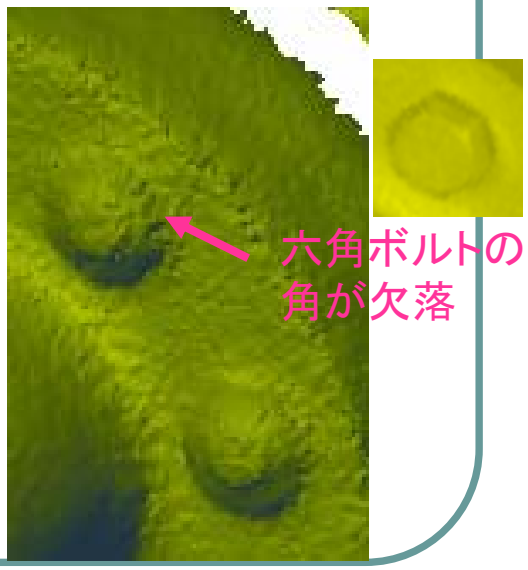
最小2乗法は，異常値に弱い ⇒ ロバスト推定

式の形が少し違うだけ

平滑化

- **ノイズの除去が不十分．角が欠落する.**





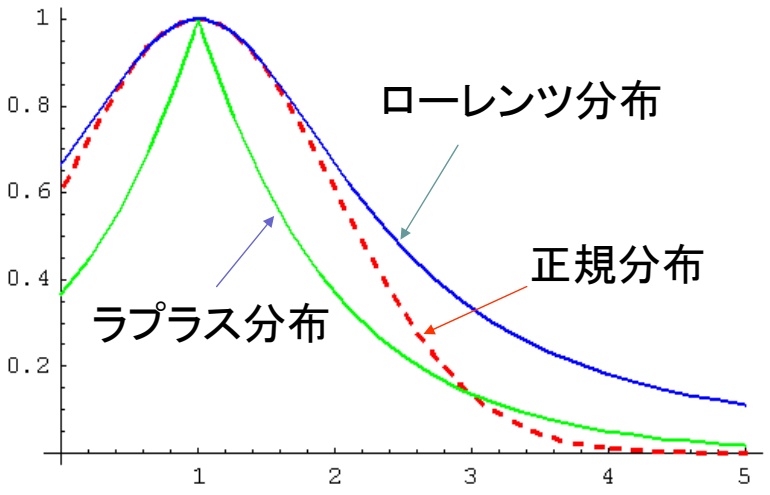
信頼できる微分値を得るために，もっと品質を上げたい.

ロバスト推定による平滑化

- ロバスト推定（最尤推定）
  - 誤差を持った計測値から，本来あるべき値を確率的に推定する.
    - 計測値は，誤差を持つ.
    - 元はどんな曲面に乗っていた可能性が高いか？

⇒　誤差分布によって答えが異なる.

誤差分布のモデルの例



平滑化の式（移動最小二乗法）

- **正規分布誤差を仮定**

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}} \left(\sum_{j \in N_i} r_i^2 \right)$$

確率最大となる
曲面式の算出式

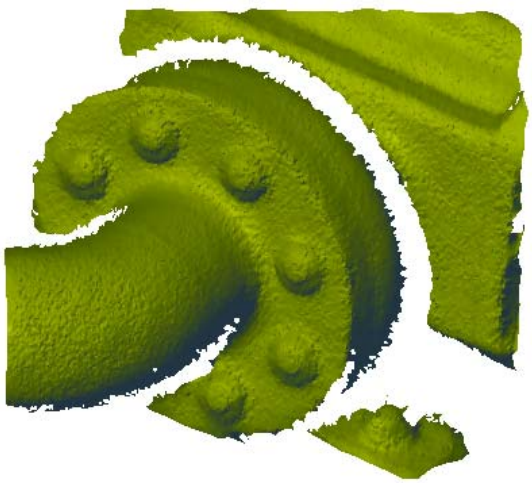
二乗誤差を最小化

- **ローレンツ分布誤差を仮定**

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}} \left(\sum_{j \in N_i} \log(1 + r_i^2) \right)$$

単純な変更だが，異常値が多いと結果は大きく異なる．

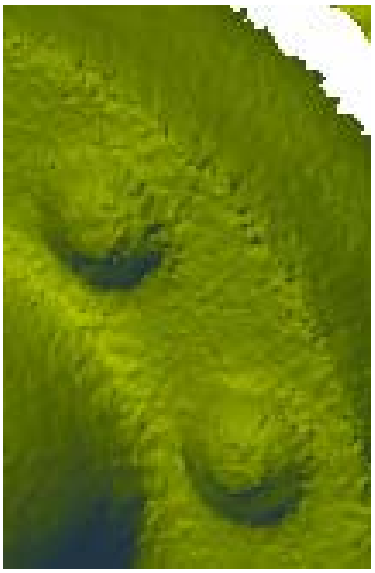
誤差と異常値に強い平滑化



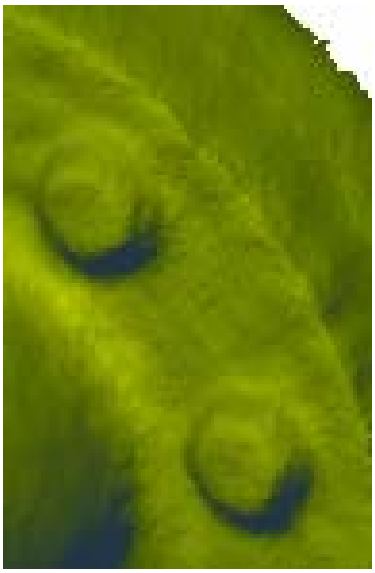
正規分布を仮定

ローレンツ分布を仮定

誤差と異常値に強い平滑化

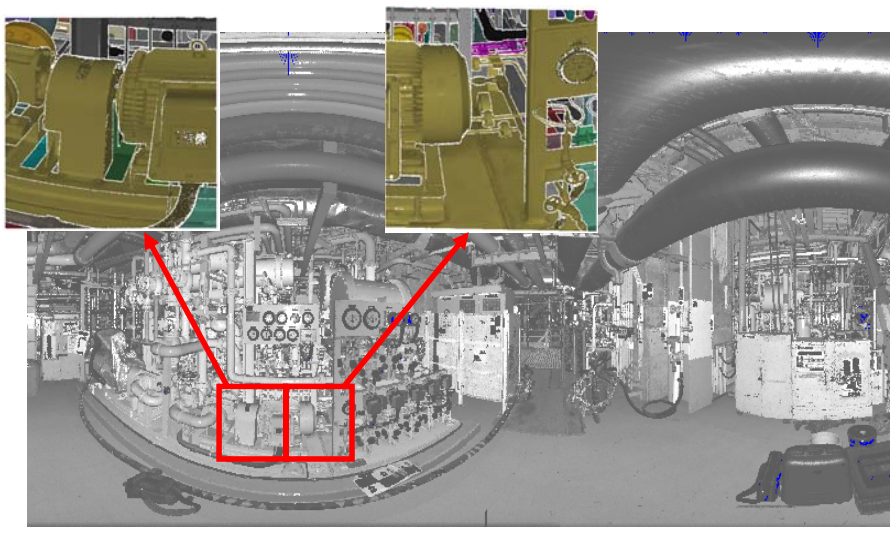


正規分布を仮定



ローレンツ分布を仮定

大規模データの処理

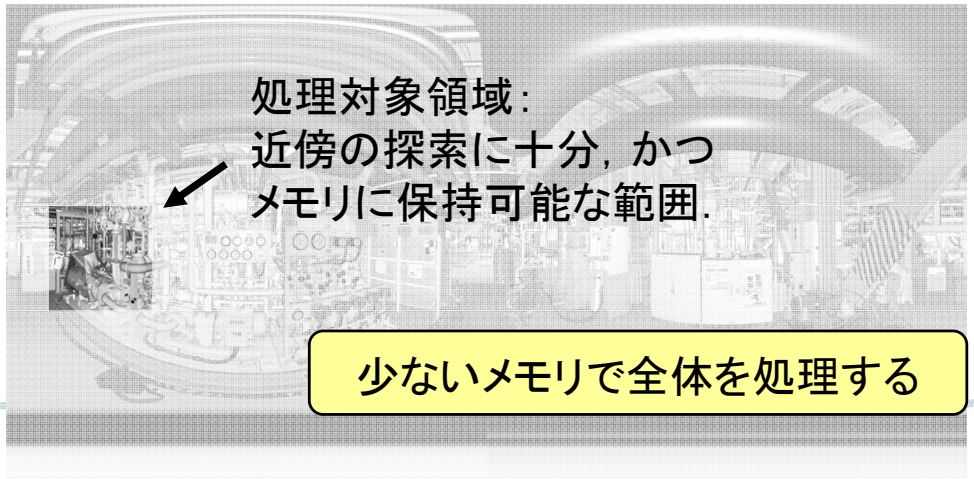


全体を一括して処理するには
計算コスト，メモリ容量の制約

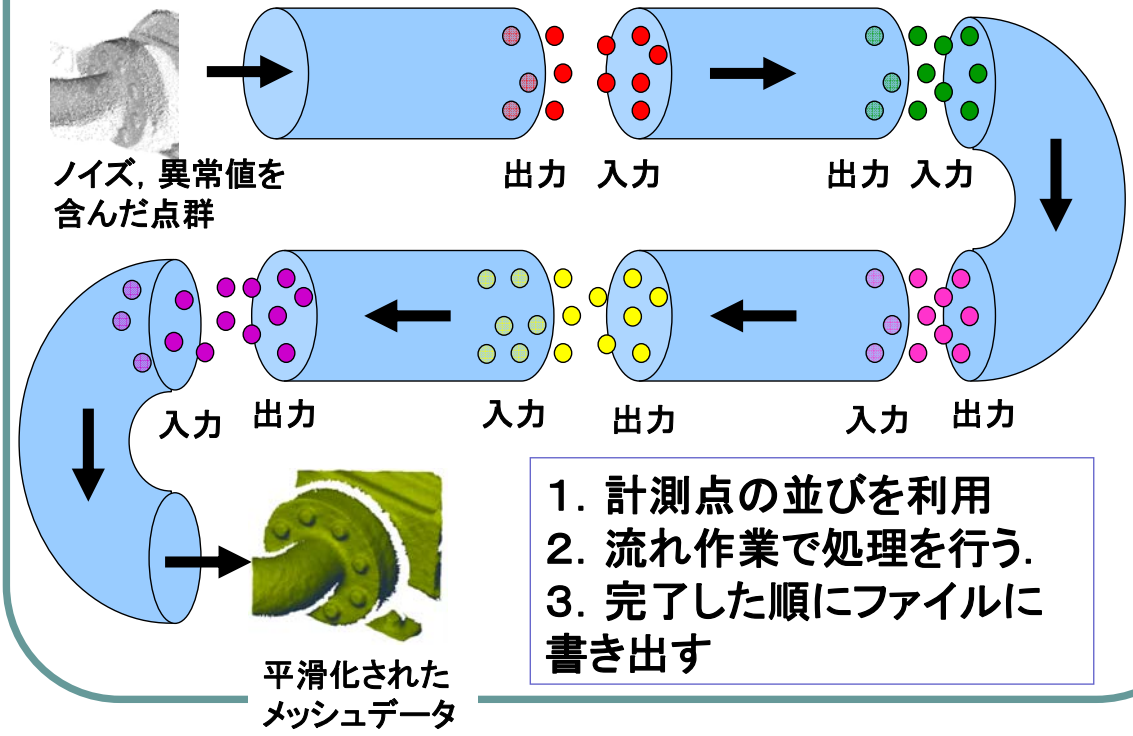
全体を64分割しても
100万頂点弱.

ストリーミング処理

- 限られた空間のみをメモリに保持.
- HDDをランダムアクセスしない.
- データは掛け流し方式.



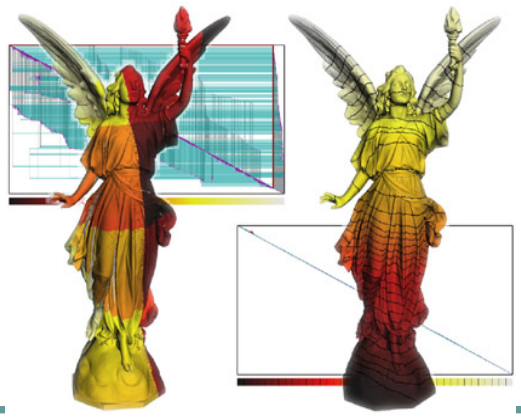
パイプライン処理



1. 計測点の並びを利用
2. 流れ作業で処理を行う.
3. 完了した順にファイルに書き出す

Streaming Mesh [Isenburg 2005]

- 実メモリ量を超える巨大なメッシュモデルを逐次解釈するためのファイルフォーマット
  - 頂点、面情報の混在
  - 古い頂点をメモリより開放



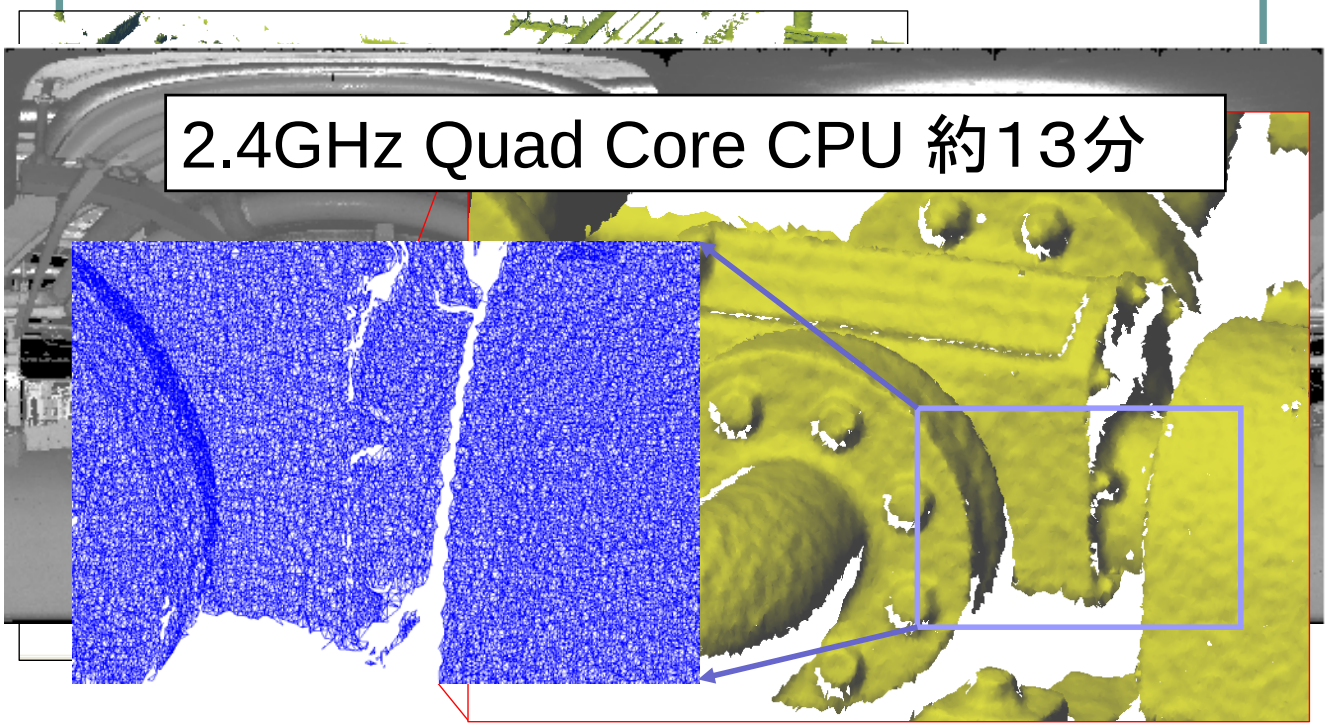
model.sma

```
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)
⋮
```


Streaming 型のMesh生成

- **平滑化によって、メッシュが裏返ることがある。**
- Streaming Delaunay Triangulation
  - **順序が多少入れ替わっても、正しく三角形分割する。**

平滑化メッシュ生成結果



セグメンテーション

多数の部材が混在

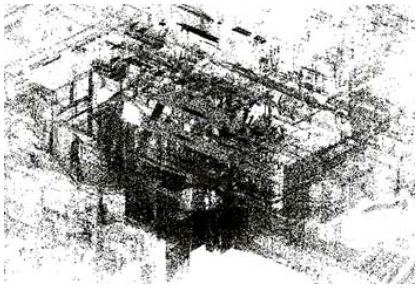
- 点群の切り分けが必要．混ざったものを分けるのは大変！
- 多様な形状，多様なスケールの部材が混在する．それぞれ，数cmから数m．

⇒ 複数に分ける

連結性

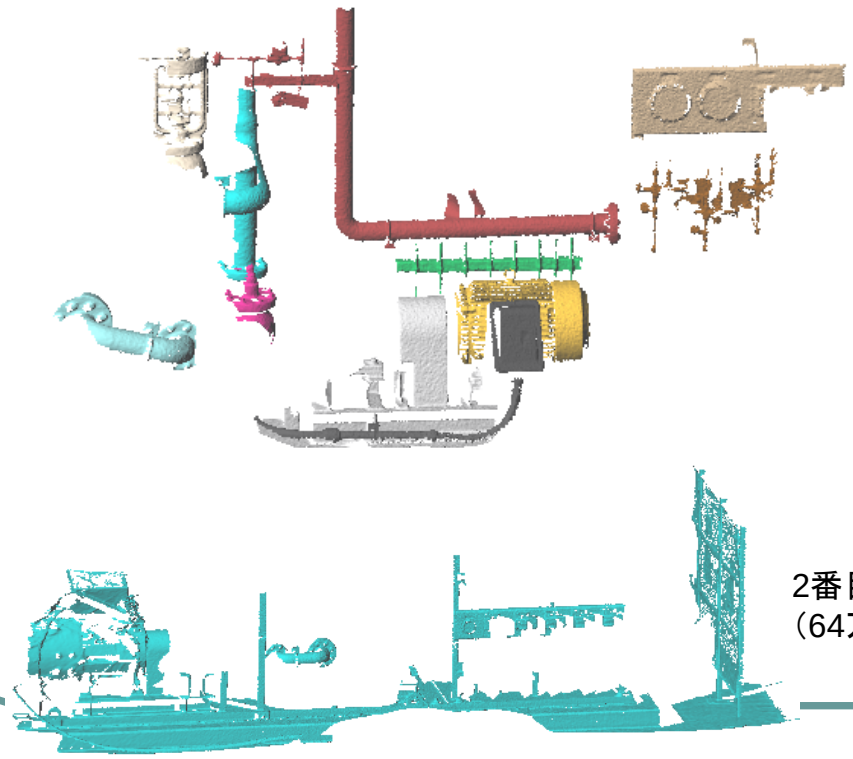
曲率、平面成分

局所処理と全体最適



データは点の集まり．
各点は様々な部材に由来する．

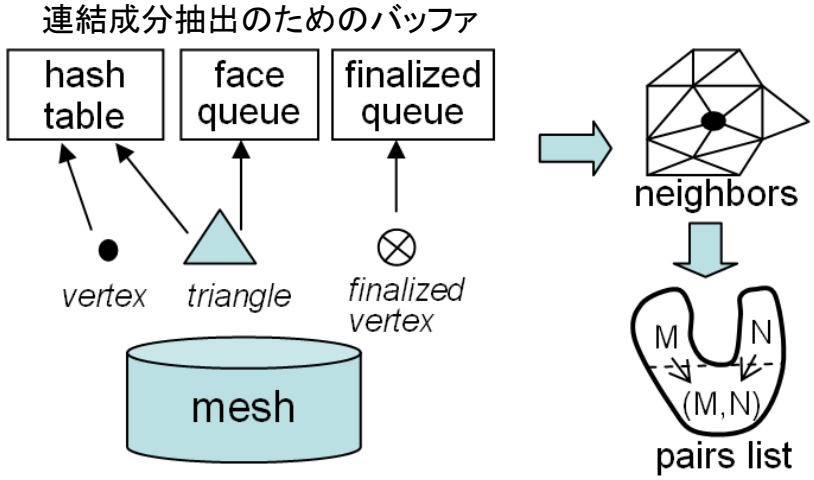
連結成分



2番目に大きい領域
（64万6181頂点）

ストリーミング処理による連結成分の抽出

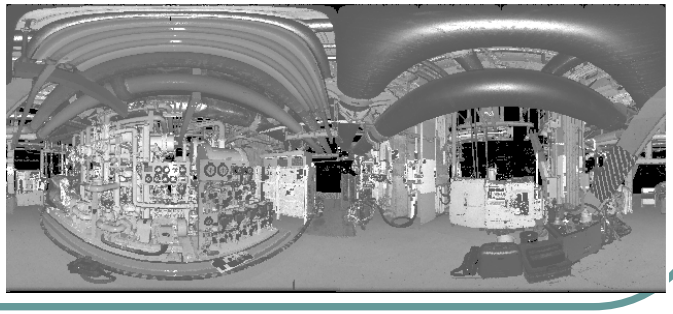
- 1回目のスキャン：異なる番号の領域をマージする必要があるときは，その領域番号のペアを保存する．
- 2回目のスキャン：頂点と三角形のデータを領域番号に応じたファイルに書き出す．



大規模点群からの連結成分の抽出

- 5000万頂点⇒平滑化メッシュ(簡略化)
- 連結領域数
  - 4229領域 (領域あたりの平均頂点数 1768)

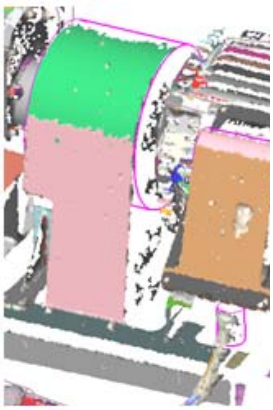
  (ただし, 25頂点以下の領域はカット(5万9517 頂点))
- 最大領域
  - 255万5402頂点

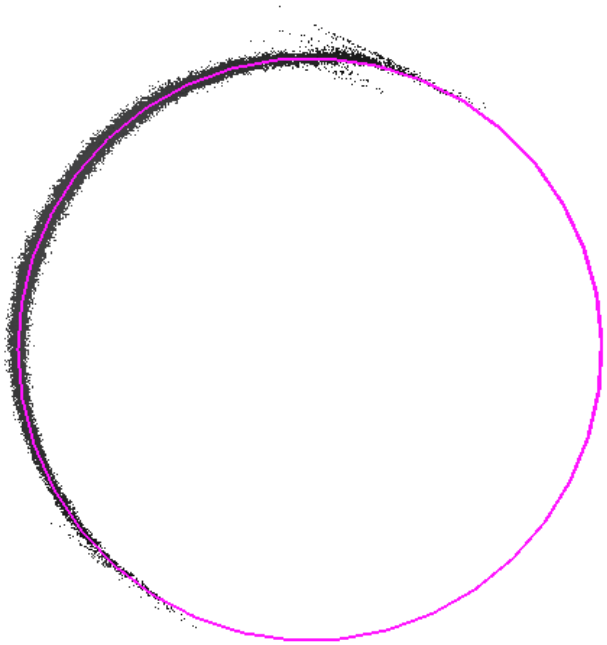


形状認識、形状マッチング

- 研究分野
  - Change Detection
  - Geometry Search
- 手法
  - 統計量
  - 特徴量
  - 2次元投影
    - スピンイメージ、輪郭
- 課題
  - 部分形状の扱い
  - 大規模点群の扱い

幾何曲面当てはめ





非線形最適化による曲面当てはめ

自由度

平面	球面	円筒面	円錐面	トーラス
3	4	5	6	7

Failthful Least Squares Fitting (Lukacs, et .al.)

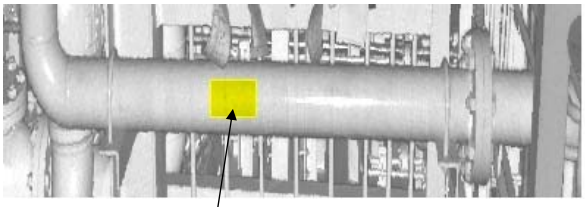
$$d(\mathbf{a},\mathbf{p}_i)=\sqrt{g(\mathbf{a},\mathbf{p}_i)}-h(\mathbf{a},\mathbf{p}_i)$$

↓ 修正距離関数の最小化

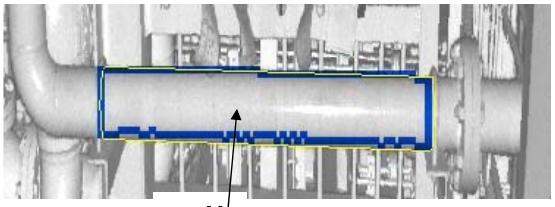
$$\sum \tilde{d}(\mathbf{s},\mathbf{p}_i)^2=\sum\{(g-h^2)/2h\}^2$$

領域成長法による曲面フィッティング

- ユーザからは透視投影画像のみが見える
  - 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.
  - 曲面式の計算.



シード領域



配管

配管のソリッドモデル

並列処理

- GPU処理
  - 高性能グラフィックス処理 （超並列）
  - CUDAなどの汎用(？)言語
  - 圧倒的な高性能
  - ただし、プログラミングは難しく、それだけで研究になる。

さいごに

- 技術的な裾野が広い。
  - 単独でカバーするのは無理。
- 研究者の参入がかかせない。
  - 産学連携。
  - 研究者が研究テーマに選びやすい環境整備。
    - スキャナが高価　→　標準データ
  - 検証データの整備。