

コンピュータ工学特別研究報告書

題目 3次元点群からの3Dモデル生成
-点群再標本化による方法-

学生証番号 044156

氏名 大平 貴達

提出日 平成27年1月23日

指導教員 蚊野 浩

京都産業大学
コンピュータ工学部

要約

Kinect で物体形状を入力した 3 次元点群にノイズ除去や位置合わせなどの前処理を施すことで、物体の全体形状を表す 3 次元点群データを得ることができ、本研究の目的は、この 3 次元点群データを三角パッチのメッシュで表した 3D モデルに変換し、3D プリンタで出力することである。

予備実験で、PCL ライブラリに実装されている関数を用いてメッシュ生成を行った。そのメッシュには不規則な穴が生じたり、ポリゴンが交差するという問題が発生し、3D プリントすることができなかった。これは、位置合わせなどの前処理を行った結果、位置合わせ誤差や重なり部分の存在によって、点群が不規則で粗密が生じたことが原因である。そこで、点群データを規則的な位置で再標本化する手法を検討した。提案する再標本化手法の基本は、点群を主軸方向に沿って均等間隔でスライスし、それを主軸周りの角度方向に均等に分割することで領域分割し、各領域内の点群座標を平均化することである。しかし、点群には粗密があるため、領域を細かく分割すると、そこに属する点が無くなる場合が生じる。そこで、細分割した場合でも平均化による再標本化が行えるように、kdtree による最近傍点探索を用いた。最近傍探索を行うことで、平均化を行うための最適な点を求めることができる。提案手法を、球状に分布する模擬データと、サンタクローズ人形を計測した実データによって検証した。その結果、これらのデータに対して、正しく動作することを確認した。そして、最終的な 3D モデルを 3D プリンタで出力した。

目次

1 章 序論	．．．	1
2 章 3次元点群処理と 3D プリンタ	．．．	3
2.1 3次元点群の入力	．．．	3
2.2 3次元点群の処理	．．．	4
2.3 3D モデルの生成	．．．	4
2.4 3D プリント出力	．．．	5
3 章 規則的なメッシュの生成手法	．．．	7
3.1 規則的なメッシュを生成する単純な方法	．．．	7
3.2 単純な方法の問題点と提案手法	．．．	9
3.3 kd 木を用いた最近傍探索	．．．	9
3.4 メッシュの生成と 3D プリント	．．．	11
4 章 実験結果と考察	．．．	12
4.1 模擬データを用いた検証	．．．	12
4.2 実データを用いた検証	．．．	14
4.3 3D プリンタによる出力	．．．	16
5 章 結論	．．．	17
5.1 成果	．．．	17
5.2 課題	．．．	17
参考文献	．．．	18
謝辞	．．．	18
付録 3D プリンタ uPrint で造形する手順	．．．	19

1 章 序論

3D モデルは、立体物の形状をコンピュータで処理可能なデジタルデータとして表現したものである。工業製品や人物形状などの 3D モデルは様々な用途で広く利用されている。工業分野では、3D-CAD(Computer Aided Design)ソフトや 3D-CG ソフトを用いて、機械部品や装置外観の設計が行われる。このように設計した 3D モデルを NC(Numerical Control)加工機械にかけることで金型を製作し、製造機械にその金型をかけることで、機械部品を量産することが一般的である。

近年では、大がかりな加工機械ではなく、3D プリンタによって 3D モデルを実物体化することも可能である。3D プリンタの技術と 3D モデル技術は密接に関係している。これらの一般的な利用法は 3D-CAD・3D-CG ソフトで 3D モデルを作成し、これを 3D プリンタで出力することである。一方、3次元形状を設計することは高度な技術で、容易に習得できるものではない。そこで、実物体の立体形状をコンピュータに入力し、参照データとして用いることで、3D モデルの設計を容易にすることが考えられる。この時に用いる、立体物の形状を入力する装置を 3D スキャナとよぶ。3D スキャナを用いた 3D モデルの設計は、実物体の 3次元形状の生データを 3次元点群として入力することから始まる。次いで、3次元点群の前処理、3D モデルの生成を行い、最後に 3D プリンタなどで出力する。

3D スキャナには様々な製品がある。その中で、Microsoft 社が 2010 年に発表した Kinect は、従来の同種の装置と比較して非常に安価である。測定精度は、1m にある被写体に対して 2.5mm 程度であるため、高精度であるとは言えない。しかし、SDK(Software Development Kit)が提供されており、PCL(Point Cloud Library)などのオープンソースソフトウェアが利用可能であるため、3D スキャナとしても利用されている。本研究では、Kinect で入力した 3次元点群を処理して 3D モデルを生成する。

3次元点群は、3次元空間座標上の点の集合である。それぞれの点は、3次元座標値とともに、色情報、法線方向などの情報を保持することができる。立体物のあらゆる表面をカバーするように 3次元点群を集めることで、立体物全体の外観形状を表現することが可能である。Kinect で立体物を計測すると、深度カメラによって立体物の位置と表面の形状を取得し、カラーカメラで物体の色を取得する。取得した 3次元点群データに対してノイズ除去やデータ削除などの前処理を施す。次いで、点群データに対して 3D モデルを生成する。点群から 3D モデルを生成する手法にドロネー三角分割や、陰関数曲面フィッティングといった手法が存在する。本研究では、ランダムに計測される 3次元点群を規則

的な構造に再標本化した後、三角パッチを生成する手法を提案する。

今回の研究は、Kinectのような一般的に普及している3D入力装置を用いて3Dモデルを生成し、それを3Dプリンタで出力することによって、比較的容易に3Dコピーが実現可能であることを実証することが主な目的である。以下、2章で3次元点群と3Dプリンタについて述べ、3章で提案手法である3次元点群からの3Dモデル生成について述べる。そして4章で実験結果と考察について述べ、最後に5章で結論を述べる。

2 章 3次元点群処理と3Dプリンタ

2.1 3次元点群の入力

3次元点群は、3次元空間上の点の集合である。それぞれの点は、空間上の位置を表す3次元座標値を持ち、RGB値や濃淡値、法線方向の情報を与えることもできる。

本研究では点群の入力に Kinect を用いる。Kinect は、Microsoft 社が 2010 年に製品化したゲーム機用のヒューマンインターフェースデバイスである。RGB 画像を取得するカラーカメラと、被写体までの距離を測定する赤外線レーザー式深度カメラを搭載している。これらの機能を用いて立体物を撮影することによって、物体の表面を表す色や形を 3次元点群データとして取得する。我々が開発したシステムでは、Kinect が測定した 3次元データを、PCL(Point Cloud Library)がサポートする pcd フォーマットを用いてファイル保存する。

物体の全体形状を 3次元点群として取得するには、物体のすべての表面をカバーするように点群を入力する必要がある。Kinect による一回の撮影では、その位置から観察できる表面だけが点群化される。そのため、複数の視点から測定を行い、複数の点群データを統合しなければならない。図 2.1 に示すサンタクロース人形の場合、Kinect の位置を固定し、1回の撮影ごとに人形を 30° 回転させて撮影することを繰り返した。物体を 360° 回転させながら撮影した複数個の 3次元データを統合した上で、3D モデル生成のための入力データとして使用した。



図 2.1 Kinect で撮影した立体物と 3次元点群データ

2.2 3次元点群の処理

3次元点群データを処理するために、PCL(Point Cloud Library)を用いる。PCLは3次元点群データを扱うオープンソースソフトウェアで、様々なアルゴリズムが提供されている。

Kinectで取得した生の3次元点群データは、3Dモデルを生成する立体物以外のデータやノイズを含んでいる。そこで、3Dモデルを生成する立体物に対応した点群データを切り出した後、PCLの外れ値除去アルゴリズムを用いることで、立体物から大きく離れたノイズデータを除去する。

次にそれぞれの角度から撮影した3次元点群データを統合するため、PCLのレジストレーションモジュールであるSAC-IA(Sample Consensus Alignment)アルゴリズムを用いて、隣り合う3次元点群同士を簡易的に繋ぎあわせる。続いてICP(Iterative Closest Point)アルゴリズムを用いて、3次元点群データの精密位置合わせを行う。これによりそれぞれの3次元点群データ間の重なりを検出する。これを繰り返し行うことで、全体形状を表現する点群データを生成する。しかし、実際には、このような自動的な処理だけでは、複数の点群を統合することが難しかった。そこで、重ね合わせのパラメータを手作業で設定し、サンタクロースの全体形状を表す点群データを生成した。

位置合わせによって1つに統合された3次元点群データだが、それぞれの繋ぎ目部分で測定データが2重になったり、ずれたりといった不具合が新たに発生する。そこでPCLのサーフェスマジュールであるMLS(Moving Least Squares、移動最小2乗近似)アルゴリズムを用いて、3次元点群データの平滑化処理を行う。これにより、雑音の大きい3次元点群データの表面をスムーズにし、滑らかな表面に調整する。

2.3 3Dモデルの生成

今回の3Dモデルは、3次元点群データに対してメッシュを貼ることで生成する。メッシュとは多数の多角形で3Dモデルを表したものである。3Dプリンタで出力可能なメッシュの条件について予備実験を行ったところ、面に不自然な穴が無いこと、面の交差がないこと、面同士が近接して重ならないことなどを満たし、閉包状のメッシュでなければならないことが分かった。

3次元点群データにメッシュを貼る基本的な考え方は、近接する点を線分で結んで三角パッチを生成し、三角パッチで構成される3Dモデルを生成することである。3次元点群をメッシュ化する方法には、大別すると次の4種類の方法がある。

① 点群の構造を利用する方法: Kinectによる1回の測定で得る点群は、一種の

画像データであり、点群が縦横に規則的に並んでいる。このような規則的な構造を持つ点群は、その構造を利用することで、メッシュ化が容易である。ただし、複数の視点からのメッシュを統合するには、特別な処理が必要である。

- ② 不規則な点群を規則的な構造の点群に再標本化する方法: 今回のように、複数の点群を合成して得る点群は、全体として点群の並びが不規則になる。このように不規則な構造を持つ点群を、メッシュ化が容易な構造に、再標本化する。
- ③ ドロネー図を用いる手法: 2次元状に分布する点群に対しては、ドロネーによる三角分割アルゴリズムを用いることで、最適なメッシュを生成することができる。
- ④ 陰関数をフィッティングする方法: 陰関数の計算によって表面情報を導き出す手法。滑らかな表面形状を表現できる。表面を表す陰関数を求めるには、幾何学や解析学といった複雑な計算を必要とする。

本研究では、統合された不規則な構造を持つ点群から 3D モデルを生成する必要があるため、点群を規則的な構造になるように再標本化する方法を採用した。

2.4 3D プリント出力

三角パッチで表現された 3D モデルを生成し、それを STL フォーマットで保存する。STL フォーマットは図 2.2 の形式のファイルであり、3D プリンタに立体物の形状を入力するための標準的なファイルフォーマットである。

solid NAME

facet normal nx ny nz

outer loop

vector x1 y1 z1

vector x2 y2 z2

vector x3 y3 z3

endloop

endfacet

生成する三角形の
個数分ループする

endsolid NAME

図 2.2 STL フォーマット(ASCII 形式の場合)

コンピュータ理工学部が保有する Stratasys 社の uPrint で 3D プリントする場合、STL フォーマットのファイルを、3D プリンタ付属ソフト CatalystEX で読み込む。CatalystEX は、読み込んだメッシュデータから、実物体を表現する領域と、印刷する際に型崩れを防ぐサポート部分を表現する領域を計算する。この変換は、立体物データに不適切な穴などの不具合が生じていると、正しく出力することができないため注意が必要である。最終的に印刷する際、実物体を表現する領域をモデル材によって、物体の形成を補助する領域をサポート材によって、それぞれ物体の下層から順に層構造で溶融積層することで物体を印刷する。全体を印刷した後、アルカリ性の溶剤でサポート材を除去する。

3 章 規則的なメッシュの生成方法

3.1 規則的なメッシュを生成する単純な方法

本研究で 3D モデルを生成する点群は、Kinect で多方向から撮影した複数の点群を統合して一つにしたものである。統合の処理によって、元の点群に存在した規則的な構造が失われている。また、点群の結合部分では不規則な重なりが生じる。位置合わせが不正確な部分では、点群の並びが二重になる。この状態で PCL に実装されている GreedyProjectionTriangulation 関数を用いてパッチ生成を行うと、一部で面の生成に失敗したり、面の辺同士の連結性が保てないなどの不具合が発生した。そこで、3次元点群を規則的な位置で再標本化する手法について検討した。

規則的な位置は立体物の表面に対応するので、なんらかの 2 次元座標系 (2 軸) を設定し、この 2 軸上の規則的な位置で再標本化することになる。これに 2 次元直交座標を用いると、立体物の全周をカバーするために、複数の座標系を組み合わせる必要があり、接合部分の処理が複雑である。

それに対して、第一の軸を 3 次元点群の主成分方向の軸 (主軸) とし、第二の軸を主軸まわりの回転角度にとることで、比較的容易に立体物の全体形状に対応させることができる。本研究では、このように主軸を一本決め、その軸周りに均等間隔に点群を再標本化することを基本的な考え方とする。ただし、この方法でも、複雑な立体形状に対しては、全体を正しく対応させることができない。その場合には、3次元点群を分割し、それぞれを単純な形状としてモデリングしたのち、接合部を処理する必要がある。

図 3.1 に、基本的な考え方に従って規則的な構造の 3D モデルを生成する方法を示す。まず、3次元点群を主軸方向に均等に分割する。分割された部分的な点群をスライスとよぶことにする。スライスを、スライスの中心から角度方向に均等に分割することで、領域分割する。それぞれの領域内の点群は立体物の微小表面に対応した点群である。この点群の重心 (平均) を再標本化点とする。再標本化点は上下・左右の再標本化点と連結していると考えると、容易に規則的なメッシュを生成することができる。

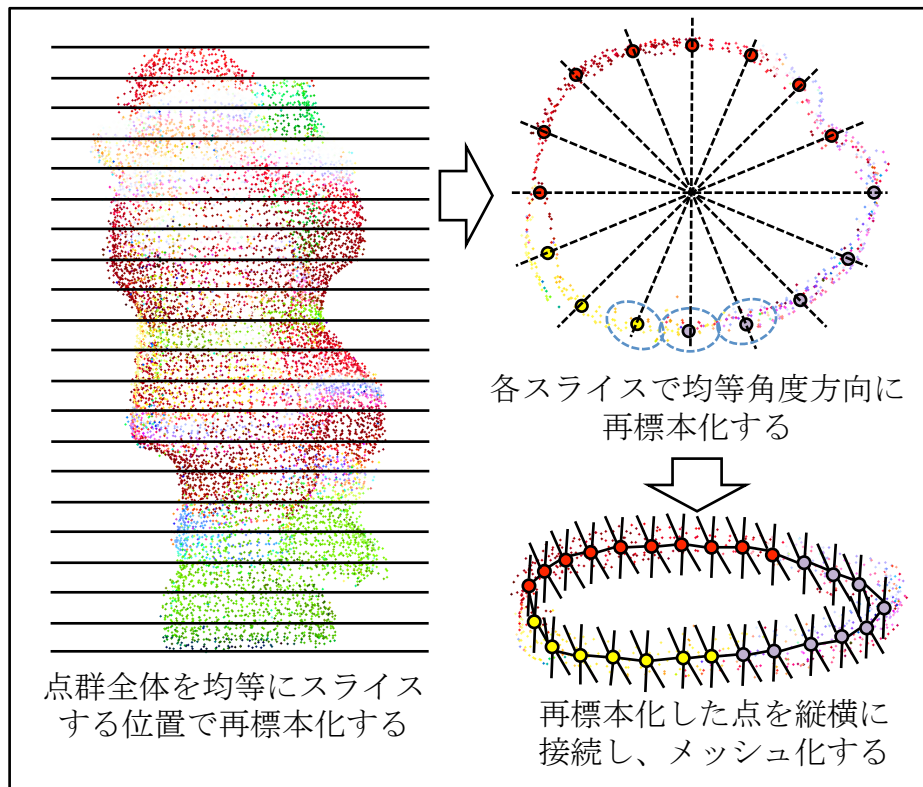


図 3.1 規則的なメッシュを生成する単純な方法

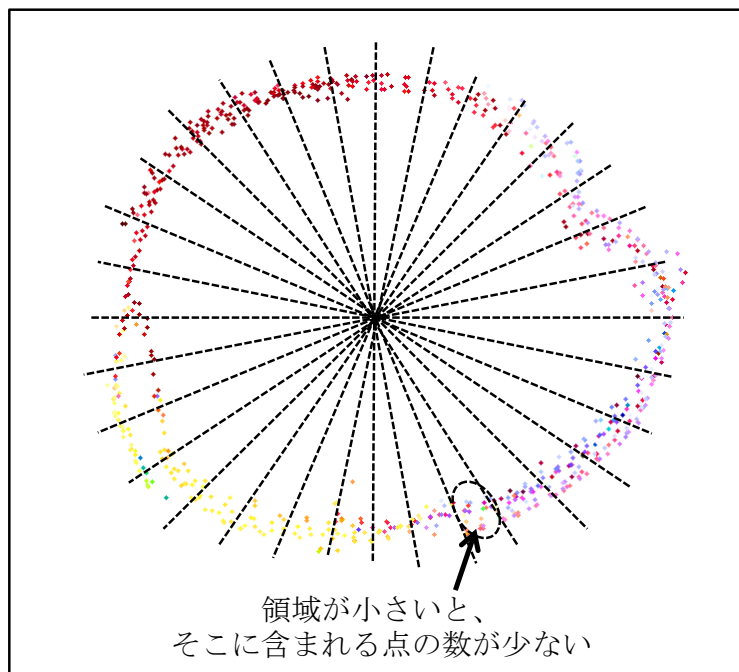


図 3.2 単純な方法の問題点

3.2 単純な方法の問題点と提案手法

単純な方法で粗い 3D モデルを作ることができた。しかし、精密なモデルを作るために、主軸方向と角度方向の分割数を細かくすると、分割された領域に含まれる点の数が少なくなる (図 3.2)。元の 3 次元点群は、点の分布に疎密があるため、それほど細かく分割していないにもかかわらず、点の数が 0 になる領域が発生した。この問題に対応するためには、点の数が 0 の領域の再標本化点を周囲の領域の点から補間しなければならない。

この問題に対して、次の方法を考案した。

- ① 規則的な位置における再標本化点を、単純な方法で、粗く生成する。
- ② 粗い再標本化点をもとに、縦横 2 倍などの位置で、高精細な再標本化点の初期値を推定する。
- ③ 推定された再標本化点の近傍で、点群を探索し、 k 個の最近傍点を検出する。
- ④ 検出された k 個の点の平均値を新たな再標本化点とする。以上。

この方法を実現するためには、与えられた 3 次元座標に対して 3 次元点群を高速に探索し、 k 個の最近傍点を検出する必要がある。次節では、それを実現する kd 木を用いる方法について説明する。

3.3 kd 木を用いた最近傍探索

最近傍探索は、距離空間に存在するデータ群に対してある位置を指定し、それも最も近いデータ、あるいは、近い順に複数のデータを探索する処理である。本研究では、最近傍探索に kd 木を用いた。kd 木は k 次元空間で 2 分木探索を再帰的に行い、データ全体を領域分割するデータ構造のことである。データ数が膨大な 3 次元点群の中から、高速に最近傍探索が可能である。

ここでは、2 次元データに対する kd 木の例と、これを用いた最近傍探索の手順を説明する。図 3.3 の 10 個の青のマークは 2 次元空間に散らばったデータの例である。これに対する kd 木を構成する。まず、 x 軸の値について、全データ点の中央値を求め (この場合 6)、2 次元空間を 2 分割する。この分割面を l_1 とする。分割面上の点は分割面よりも大きい側に属するとする。次いで、2 分割されたそれぞれのデータについて、 y 軸の値の中央値を求め、この値によってそれぞれの領域を 2 分割する。その分割面を l_2 、 l_3 とする。同様の手順を、全てデータが一つの領域に属するまで繰り返す。その結果、得られるのが、図 3.3 の $l_1 \sim l_9$ の分割面である。

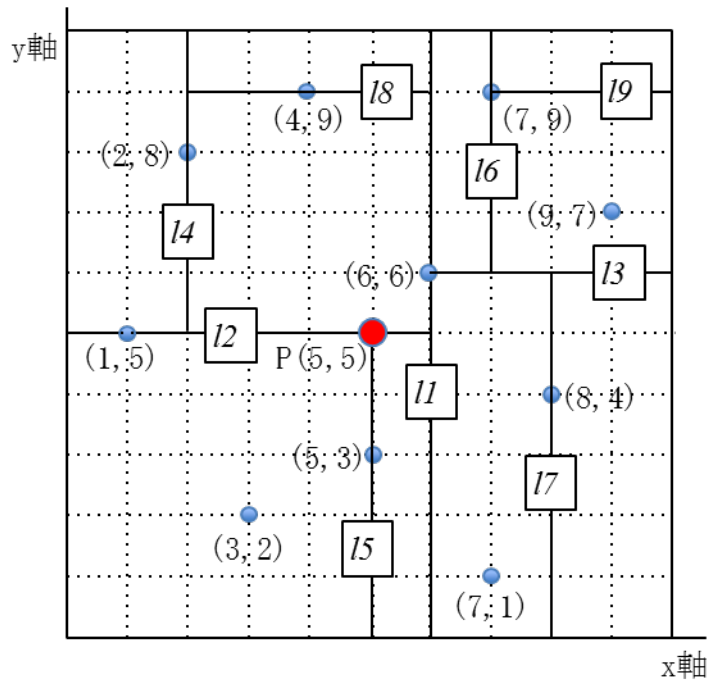


図 3.3 kd 木による空間分割の例

図 3.3 の空間分割を 2 分木で表現すると、図 3.4 になる。ここで、葉ノードはデータ点に対応し、それ以外のノードは分割面に対応する。この kd 木にクエリとして点 $P(5,5)$ を与えた場合の最近傍探索の手順は次のようになる。

- ① kd 木を深さ優先探索して葉ノードに達する。この場合、点 $(2,8)$ に到達する。この点が最近傍点である保証はない。その距離を d とする。
- ② 親ノードに上がる。 P を中心とする半径 d の円と、親ノードのもう一方のノードの領域が交わるかどうかを調べる。交わらない場合、更に親ノードに上がる。交わる場合、その葉ノードを調べ、より近い場合には d の値を更新して、更に親ノードに上がる。

この場合、点 P の最近傍点は $(6,6)$ である。この結果を得るには、ノード 11 まで遡らねばならず、それほど効率的な探索にはなっていない。しかし、kd 木による探索の計算量は、データの点数を N として $O(\log N)$ であることがわかっている。これは、全探索の計算量 $O(N)$ に比べて効率的である。

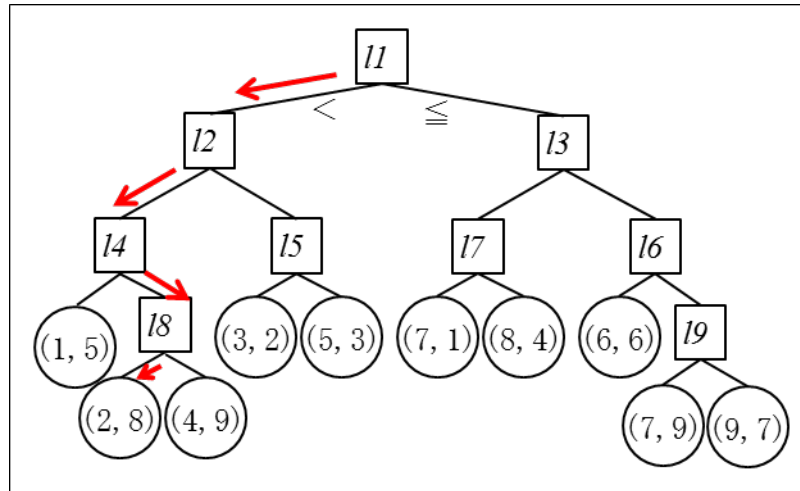


図 3.4 kd 木を用いた探索の例

3.4 メッシュの生成と 3D プリント

再標本化した点群は規則的に配列しているため、メッシュを生成することは容易である。まず、一つのスライスに属する点は角度によって、左右の隣接関係が決まる。次に、連続するスライスの点の間では、同じ角度範囲に属する点の間に上下の隣接関係が決まる。従って、上下左右の隣接 4 点が容易に決まる。その対角に適切な規則で線分を決めることで、2 つの三角パッチを生成する。この処理を全ての点に施すことで、メッシュが生成される。なお三角パッチの表側は、三角形の頂点を反時計回りに指定し、右ねじが進む側になる。

今回は Stratasys 社の uPrint で 3D プリントを行う。この 3D プリンタには、メッシュデータを 3D プリントするための造形データに変換する付属ソフトウェア CatalystEX があるためこれを利用する。生成したメッシュデータを元に 3D プリントを正しく行うためには以下の点に特に注意する必要がある。

- ① 面生成のできていない穴を無くした完全閉包なメッシュデータにする。
- ② 面同士の交差が無く、隣り合う辺同士に連結性のあるメッシュデータにする。

これらが守られていないと、3D プリントを行っても物体を正しく書き出すことができない。

4 章 実験結果と考察

4.1 模擬データを用いた検証

模擬データとして中心点 $(x,y,z)=(0,0,0)$ から距離 1 の地点にランダムに散らばる 1000 個の点群データを用いた(図 4.1)。再標本化点を計算する再標本化位置を、主軸方向のスライス数 10、角度方向の均等な分割数 16 の計 160 個の領域とした。各領域に属する点群の座標値を平均化することで、入力点群を再標本化した(図 4.2)。再標本化を行った点群データに三角パッチを用いてメッシュ生成を行った(図 4.3)。なお、図 4.1~図 4.3 は点群やメッシュの状態を確認するために用いた、フリーの 3DCG ソフト **Blender** の画面である。

このように比較的単純な球の模擬データを、点群の再標本化によってメッシュ生成することを可能にした。

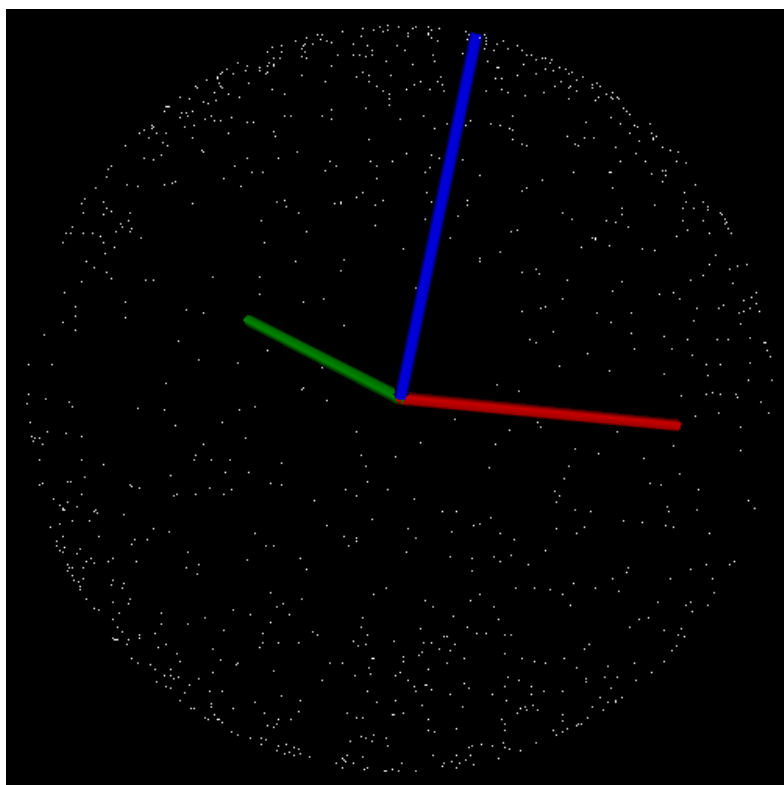


図 4.1 中心から距離 1 の地点にランダムに生成された 1000 個の点

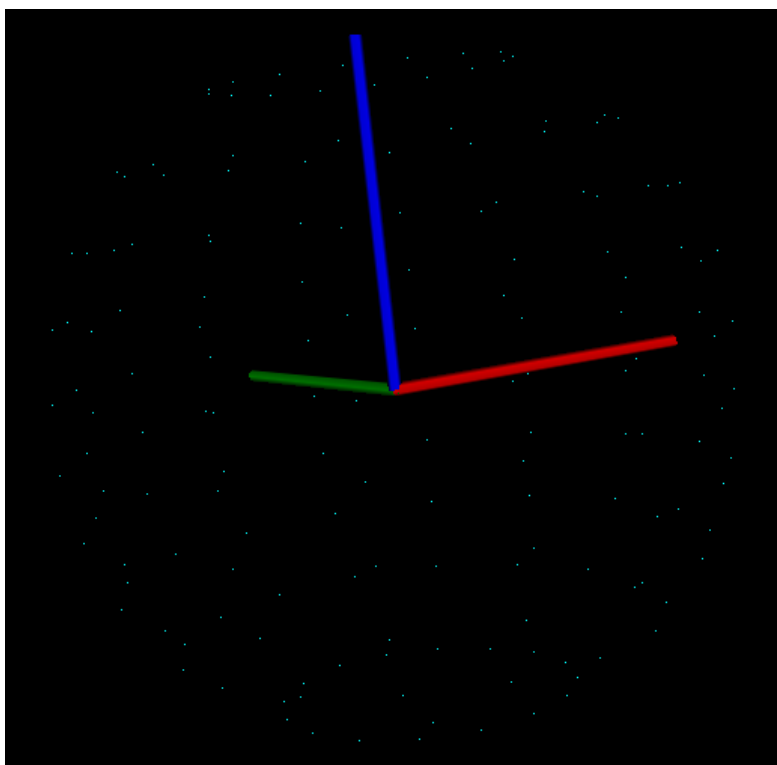


図 4.2 入力点群を再標本化した点群

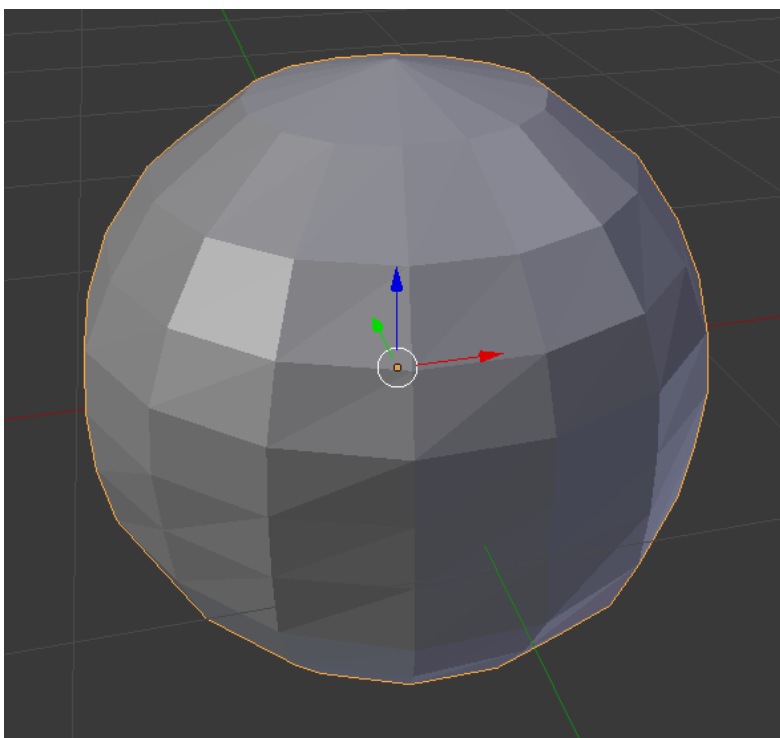


図 4.3 再標本化した点群に対するメッシュ生成

4.2 実データを用いた検証

実データとして、縦 76cm、横 35cm のプラスチック製サンタクロース人形を 3次元点群として取得し、統合、位置合わせ等の前処理を施したもの(図 4.4)を、提案手法によって 3D モデリングした。

まず点群の再標本化位置を、主軸方向のスライスを 20、角度方向に均等な分割を 32 の計 640 個の領域として、各領域に含まれる点群の座標値を平均化することで再標本化を行う(図 4.5)。ついで、再標本化の位置を縦横に 2 倍に増やし、それらの初期値を補間によって求める。そして、その初期値近傍を kd 木を用いて探索する。探索した点の座標の平均値を最終的な再標本化点とする。探索した点は、初期値近傍の 20 点である。再標本化を行った点群に対して三角パッチによるメッシュ生成を行う(図 4.6)。メッシュ生成を行ったデータを CatalystEX で読み込んで造形データに変換する(図 4.7)。図 4.7 の赤色で表示されている部分がサンタのモデルを表す部分で、紫で表示されている部分は造形時の型崩れを防ぐサポート材である。

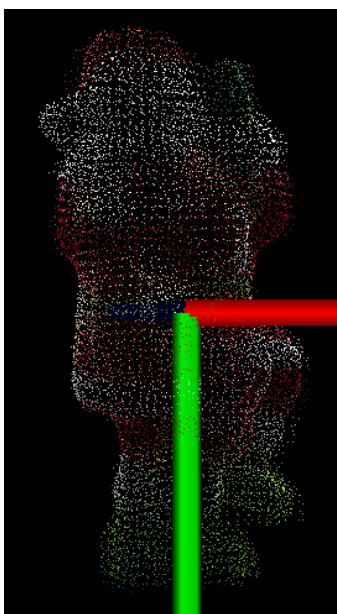


図 4.4 再標本化前の入力点群

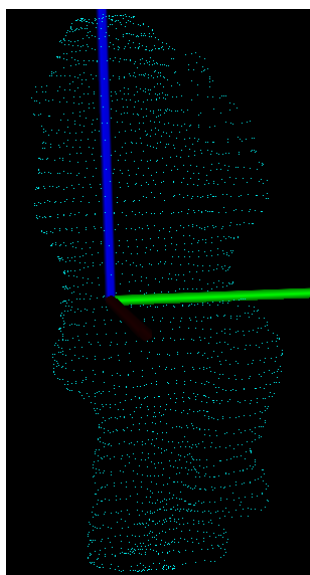


図 4.5 入力点群を再標本化した点群

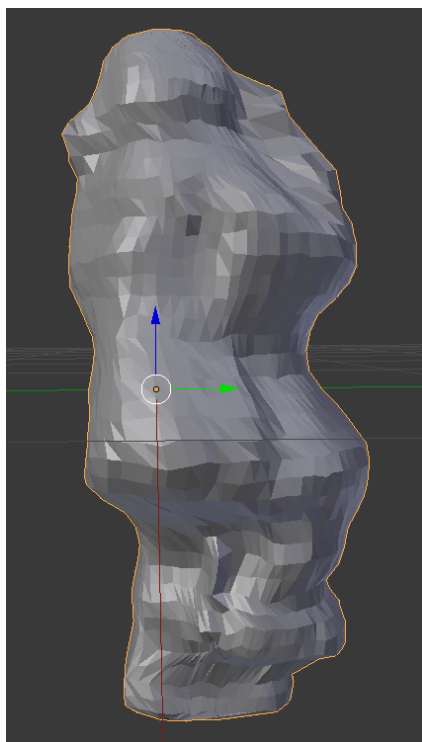


図 4.6 再標本化した点群に三角パッチでメッシュ生成

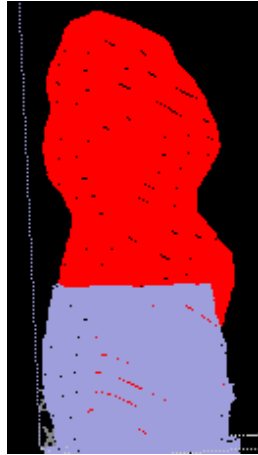


図 4.7 造形データへの変換

4.3 3D プリンタによる出力

提案手法によって造形データに変換されたデータを、実際に 3D プリンタで出力した。印刷物のサポート材を除去することで、物体の造形が完了した(図 4.8)。



図 4.8 3D プリンタによる出力

5 章 結論

5.1 成果

立体物の形状を 3 次元点群として計測し、3 次元点群処理、3D モデリング、3D プリンタによる出力という一連の処理を実現し、3D コピーシステムを開発した。本研究では、立体物の全体形状を表現する 3 次元点群から 3D プリント可能な 3D モデルを生成することに取り組んだ。当初、点群の統合の際に規則性が乱れたことによって、三角パッチの生成に不具合を生じた。そこで並びが規則的な点群になるように、点群の再標本化を行った。単純な方法では、再標本化をする領域を分割し、領域内の点群を平均化することで再標本化点を得ることができる。しかしこのままでは粗い点群になる。また領域の細分化によって領域内の点群の粗密が極端になることで、領域内での点群の平均化が正しく行えなくなる。そこで細分化した領域では、推定した再標本化点の初期値を中心に、kd 木最近傍探索を用いて平均化する点を探索することで、再標本化点を生成することができた。これで三角パッチによるメッシュ生成が可能になり、3D プリンタで印刷が可能になった。

5.2 課題

物体の表面形状の再現度はあまりよくなかった。今回の場合、サンタクロース像の大まかな形状を表現することはできたが、顔の凹凸や、服のシワなどといった細かい部分の表現は難しかった。

また、今回開発した手法では、比較的単純な物体に対する 3D モデリングしか対応することができない。例えばドーナツ状の物体に対して今回の提案手法でモデリングを行うことは困難である。これは点群の再標本化の際、主軸方向に沿ってスライスし、物体の中心から角度方向に均等に領域を分割した時、物体の表面となりうる点群の並びが複数存在するため、点群の平均化によって正しい再標本化点を求めることできないためである。

参考文献

- [1] PCL – Point Cloud Library(PCL) , <http://pointclouds.org/>
- [2] blender.org , <http://www.blender.org/>
- [3] Kinect for Windows , <http://www.microsoft.com/en-us/kinectforwindows/>

謝辞

本研究を進めるにあたり、適切な助言、機材の準備、また様々なご指導頂きました蚊野浩教授に感謝いたします。

付録

3D プリンタ uPrint で造形する手順

1. 造形物のデータを STL ファイルで準備する。
2. パソコン用のブレーカ、3D プリンタ用のブレーカが ON になっていることを確認する。
3. 3D プリンタ用のノートパソコン Think Pad を起動し、es_user でログオンする（パスワードなし）。
4. 3D プリンタの電源スイッチ（左側面）を ON。しばらくすると、材料の残量や温度が表示される。造形物を乗せるステージをセットする。
5. パソコンの Catalyst EX を起動する。STL ファイルを読み込み、パックに追加する。パソコンをイーサネットのケーブルで 3D プリンタに接続し、データを転送する。
6. 本体側にデータが転送されたことを確認し、造形開始のボタンを押す。温度が上昇し、環境温度が 75°C 程度で、キャリブレーションを開始する。キャリブレーションが終了し、材料の温度が 300°C 近くになると、プリントを始める。
7. プリント終了後、ステージを取り外す。ステージ上の造形物をスクレーパーではがす。
8. 造形物からサポート材を除去するために、アルカリ水溶液に浸す。例えば、市販の漂白剤ブリーチの原液に浸すと、12 時間程度でサポート材を手ではがすことができる程度にもろくなる。