

視点からの距離に応じて詳細度の変化する ボクセル表現を用いた VR 可視化システムの提案

大久保 桐吾¹ 伊藤 正彦¹

概要: 点群データはデータ量が膨大な上、データ量を削減し可視化すると視認性が低下するという問題がある。点群データをボクセルに変換することでデータ量を削減しつつ視認性を維持することができる。この際、注目度に応じてボクセルの詳細度を変えることでより効率的にデータ量の削減と視認性の維持を両立できると考えた。本研究では、視点の位置に近い物体を注目度が高いと考え、点群データを視点からの距離が近いほどサイズの小さなボクセルで表現する手法を開発する。実際に、複数の雪像データを VR 空間で可視化し手法の有効性を確認する。

1. はじめに

近年、様々な実世界の物体が 3 次元の点群データとして保存されるようになってきている。これらのデータを用いて VR のアプリケーションを構築することが可能となっている。

点群データはデータ量が膨大であり、VR 空間で点群の膨大なデータを読み込み表示するには時間がかかる。また、可視化した際にオブジェクトから適切な距離を取らなければ、オブジェクトの形状を認識することが難しいという問題がある。表示する点の大きさが小さいと、オブジェクトが視点から近い位置にある場合に、点と点の間に隙間ができて形状を認識しづらくなる。また、表示する点の大きさが大きいと、視点から遠い位置にある場合に点同士が重なり形状が認識できなくなるうえ、遠くから見る際は点群の詳細が認識できないため、すべての点を表示する必要はなくなる。

そこで、点群データをボクセルデータに変換することでデータ量を減らし、点ではなくキューブの形状をしたボクセルを用いることで、視認性を維持した状態で可視化できると考えた。また、その際に遠いものは粗く、近いものは細かくすることで視認性を維持したままデータ量を減らすことができると考える。

本研究では、さっぽろ雪まつりの雪像データを用いて点群データをボクセルデータに変換し、点ではなくキューブで可視化することで、データ量を減らしたものでどういった立体物かをはっきりと認識できると考えシステムを

開発し、VR デバイスでの実装を行う。

2. 関連研究

Oscar Martinez-Rubi[1] らは、点群データの巨大な多重解像度データ構造を作成するアルゴリズムを発表した。このアルゴリズムでは、点群内に存在する点同士の距離を計算によって求め、Octree の深さとしてユーザーが設定した様々なレベルの Octree ノードに定義する。この処理をすべての点がノードに追加されるか、Octree の深さを超えるまで繰り返す。このとき、Octree の深さを超えた点は破棄され、どのノードにも追加されない。

視点からの距離に応じて段階的に Octree のレベルを変えていくことで、距離に応じて可視化する点の大きさを変化させる。距離が近ければ高解像度のデータ、遠ければ低解像度のデータとなる。

本研究と関連研究の違いとして、可視化するオブジェクトが点ではなくボクセルであるということ、また VR デバイスを用いて可視化するという点、また、現在はボクセルのサイズ変化の際に Octree を利用せず、任意に設定した数値を用いるという点である。

3. 提案手法

3.1 データについて

雪像作品の点群データは、2024 年に開催された第 74 回 さっぽろ雪まつりのものを用いた。作品を様々な角度から複数枚撮影を行い、撮影した写真と Luma AI^{*1} を使用し、点群データを作成した。

¹ 北海道情報大学 情報メディア学部

^{*1} <https://lumalabs.ai/dream-machine>

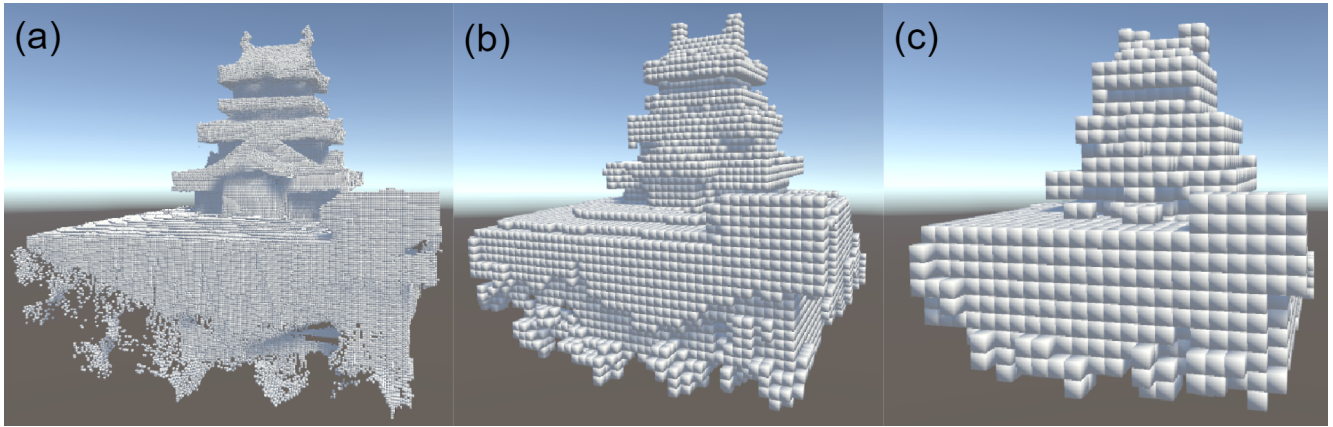


図 1 (a) ボクセルサイズ 0.01 (b) ボクセルサイズ 0.05 (c) ボクセルサイズ 0.1

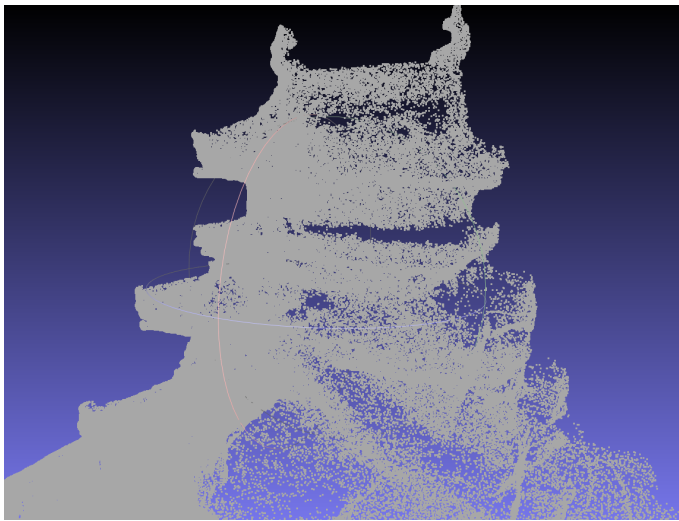


図 2 城の点群データの可視化結果

3.2 点群データをボクセルデータに変換

城の雪像データから生成した点群データ(図2)をボクセルデータに変換した。データの変換には、Google Colaboratory 内で Open3D というライブラリを用いて作成したシステムを使用する。点群データをボクセルデータに変換する方法として、三次元空間に配置された点を一辺の長さが任意の大きさをした立方体のグリッドで囲み、グリッド内に既に点があればグリッド内のすべての点を削除する。グリッドの中心点のみを表示することで、データ数を減らすことができる。

点群データをボクセルデータに変換する際、点群データの際のデータ数は 702270 データであるが、図 1(a) では 110660 データ、図 1(b) では 6673 データ、図 1(c) では 1835 データといったようにデータ数が削減できる。この時、図 1(a) のデータが約 1/6 程にしかデータが削減できていないためあまり軽量化はできていない。一方、図 1(b) は約 1/105、図 1(c) は約 1/382 とデータ量の大幅な削減ができています。

3.3 ボクセルデータをキューブで可視化

視点からの距離に応じて詳細度の変化するボクセルを Unity 上で可視化した(図1)。ボクセル一つ一つをオブジェクトとして生成するのではなく、ボクセルによって構成される立体物全体を一つのオブジェクトとして可視化した。現在は、ボクセルデータの可視化に、点の座標を中心としてキューブ形状のメッシュを生成する方法を用いているが、将来的にはシェーダーを利用して可視化する方法に変更したいと考えている。

視点からの距離に応じて可視化を行う際に、視点からどれだけ離れたらボクセルのサイズを変化させるかは、任意の距離を数値で指定する。また、数値で指定した任意の距離に合わせて点群データをボクセルデータに変換する必要がある。視点移動をした際は、オブジェクトに近寄れば図 1(a) のように細かく、オブジェクトから離れれば図 1(c) のように粗く、段階を踏んで変化する(図3)。

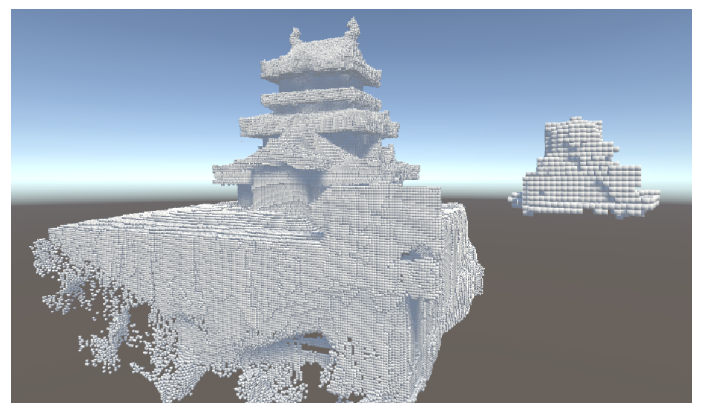


図 3 近くの細かいボクセルと遠くの粗いボクセルの比較

その結果、データ全体を一つのオブジェクトとして可視化した場合、ボクセル一つ一つをそれぞれ一つのオブジェクトと比べて、視認性に大きな変化は見られなかったが視点移動やオブジェクトの移動の際の動作の軽量化には効果があった。

4. おわりに

本研究では、視点からの距離に応じて詳細度の変化するボクセルの可視化システムの開発を行った。視点からの距離に応じて詳細度を変化させることにより、視認性を維持することとデータの軽量化に繋がるということが確認できた。可視化の際にボクセルの詳細度を変化させる適切な距離の数値と、適切なボクセルサイズを求めることができていないため、それらの数値を求めるための評価実験を行っていく。

参考文献

- [1] Martinez-Rubi, O., Verhoeven, S., van Meersbergen, M., Markus Schütz, van Oosterom, P., Alves, R. G., Tijssen, T.: Taming the beast: Free and open-source massive point cloud web visualization, *Capturing Reality 2015* (2015).