J-035

VR 空間に描いた線画像による 3D モデル検索
3D model search using line images drawnin VR space

上馬 拓己† 新田 藍花† 中井 満†
Takumi Joba Aika Nitta Mituru Nakai

1. はじめに

近年，3D カタログや Augmented Reality（AR）コンテンツなど，3D モデルが様々な分野で使われている．それに伴い，大量の 3D データを収集することも容易になっている．また，3D データファイルの検索では，キーワード検索が主流である．あらかじめデータの特徴を表すキーワードを付けておく必要があり，ファイルが多くなるほどコストがかかってしまう．そこで本研究は，VR 空間で簡単に描いた線画像で 3D モデルを検索できるシステムを開発することを目的とした．

2. 3D オブジェクトデータ

学習用データとして，Princeton 大学が公開している ModelNet [1]の 40 種類のクラスのうち，chair クラスのデータ全 989 個を用いる．検索データベース（Target）として，学習用データのうちの 50 個を使用する．また，検索用データ（Query）として Oculus Quest のコントローラで描いた 3D 線画像を用いる．図 1 は 3D モデルと 3D 線画像の一例である．また，図 1 のメッシュデータをボクセル化したものを図 2 に示す．

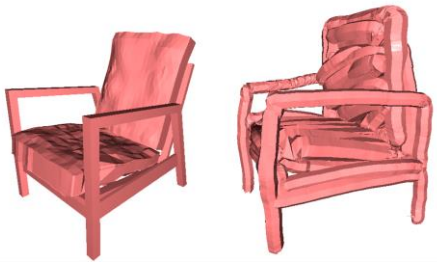

図 1　3D モデル(Target)と 3D 線画像(Query)

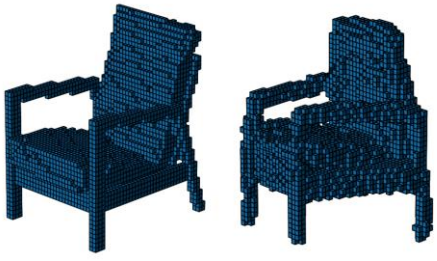

図 2　ボクセル（左：Target，右：Query）

† 富山県立大学　Toyama Prefectural University

3. 3D モデル検索システム

3.1 検索システムの構成

Target と Query を比較するために，両者をボクセルデータに変換し，オートエンコーダで特徴抽出する．具体的には解像度 48^3 のボクセルデータを入力として $6^3 \times 64$ 次元の特徴量を得る．一般にオートエンコーダの学習は入力データをエンコードして次元圧縮し，その圧縮したデータからデコードして元の入力データを再生成するようにネットワークを繰り返し更新する．本検索システムでは，あらかじめすべての Target を特徴量に変換しておき，検索のときは Query のみを特徴量に変換する．そして，Query との特徴量間の距離が最小になる Target を検索結果とする．

3.2 特徴量

本研究では，3D オートエンコーダを用いて，ボクセルから特徴抽出を行う．3D オートエンコーダは図 4 に示すように，エンコード過程では入力されたデータをキューブ状のフィルタで畳み込み，プーリングして，多層で段階的に次元圧縮する．デコード過程では圧縮したデータから再び元のサイズのデータを復元処理する．したがって，エンコードで圧縮したデータは 3D 形状の復元に必要な特徴量である．また，Query の 3D 線画像は手描きであるため，図 2 のように Query は Target に比べてボクセルが凸凹になっている．そのためノイズを付加したデータ拡張を行い，それらを復元するオートエンコーダを学習する．

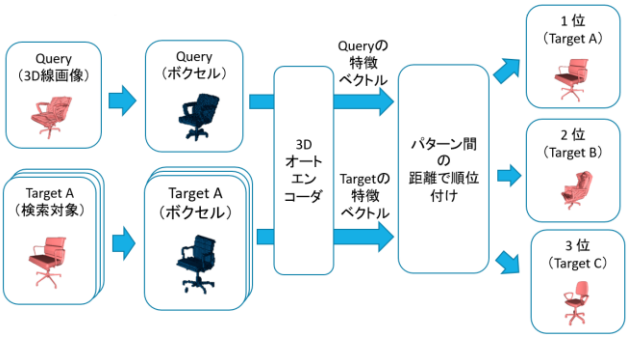

図 3　3D モデル検索システムの構成

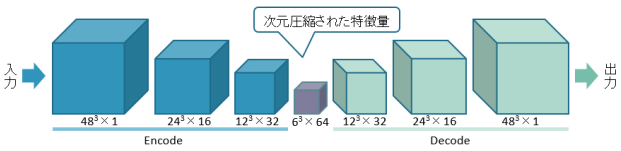

図 4　3D オートエンコーダ

4. 検索実験

4.1 実験条件

ModelNet のうち，chair クラスのサンプル 989 個とそれにノイズを付加したものを用いて，オートエンコーダを学習した．また，検索対象には，類似したモデルを除いた 50 個のモデルを使用し，この 50 個のモデルを真似て描いた線画像を Query とした．なお，この 50 個のモデルは k-means 法によって選定した．検索順位は特徴量間の距離の小さい順とし，学習の条件は以下の 3 通りとした．

(a) 実験 F3 : エンコーダ 3 層，データ拡張なし
(b) 実験 F3+ : エンコーダ 3 層，データ拡張あり
(c) 実験 F4+ : エンコーダ 4 層，データ拡張あり

また，比較として，ボクセルを用いた検索実験（実験 V ）を行った．解像度 48^3 のボクセルを，3 軸方向に平行移動して距離が最小になるように位置合わせを行った．また，点群を用いた検索実験では，Query の点群各点に対応する検索対象 3D モデルの最近傍点を求め，その点群間距離を用いた．なお，位置合わせを行わずに検索する実験（実験 P）と，回転位置合わせを行って検索する実験（実験 P+）を行った．位置合わせをするために Iterative Closest Point（ICP）アルゴリズムを用いる．ICP は，点群間距離を最小化するように回転・平行移動・スケール変換を収束するまで繰り返す．回転位置合わせ前と後は図 5 のようになる．検索実験の OS は Windows Subsystem for Linux 2 の Ubuntu18.04，CPU は Intel Core i7-10700K ，言語は Python 3.6 を使用した．距離計算には Numpy 1.19.5 の関数 norm を使用した．

4.2 評価実験

検索システムの評価として，ROC 曲線と F 値，5 位以内に得られた Target の数を調べた．ROC 曲線は検索閾値を変えたときの False Positive（FP）の割合と True Positive（TP）の割合の関係を表した曲線である．ここで TP は，Target を正しく検索できた場合であり，50 通りある．FP は誤って検索した場合であり，2450 通りある．F 値は，適合率と再現率の調和平均である．ここで，適合率は検索結果に占める正しい Target の割合であり，再現率は検索したい Target に占める検索できた Target の割合である．

ROC 曲線を図 6 に，F 値を表 1 に，Target の平均順位を表 2 に，5 位以内に得られた Target の数を表 3 に示す．

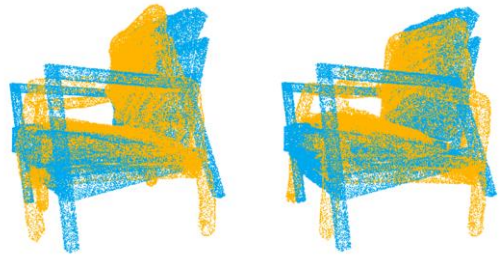

図5　2つの点群（左：位置合わせ前，右：位置合わせ後）

表 1　F 値

実験	F3	F3+	F4+	V	P	P+
F 値	0.814	0.808	0.808	0.706	0.723	0.786

表 2　Target の平均順位

実験	F3	F3+	F4+	V	P	P+
F 値	8.0	7.6	7.7	15.1	12.0	8.9

表 3　検索成功率 (5 位以内)

実験	F3	F3+	F4+	V	P	P+
F 値	62%	60%	60%	36%	16%	54%

表 4　処理時間

実験	F3+	F4+	V	P+
処理時間	0.495ms	0.326ms	2420ms	数分

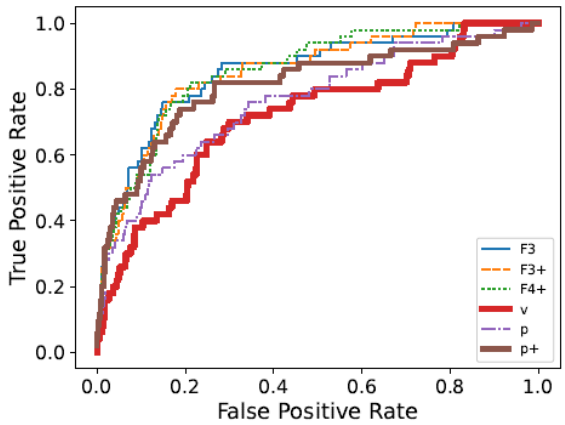

図 6　ROC 曲線

4.3 考察

いずれの実験も F3，F3+，F4+が比較的良い結果となったことから，特徴量を用いた検索が有効であると分かった．また，F3 の方が F3+ に比べて F 値と検索成功率は若干良い結果となったが，Target の平均順位では F3+ の方が F3 よりも若干良い結果となったことから，今回のデータ拡張の方法は効果がなかったと考える．また，V や P よりも P+ の方が良い結果となったことから，回転位置合わせが有効であることがわかる．

5. まとめ

3D 線画像を用いた 3D モデルの検索法として，特徴量，ボクセル，点群による検索を行い，特徴量を用いた検索によって他の手法よりも良い結果を得ることができた．処理時間について，大規模なデータベースの検索には特徴量を用いることで，大幅に時間短縮できることがわかった．今後，向きを気にせずに Query を描けるように，拡張データに回転や平行移動を加え，位置ずれや向きの違いにも強い特徴抽出器を作成する必要があると考える．

謝辞

本研究は JSPS 科研費 17K00275 の助成を受けて行った．

参考文献

[1] Z. Wu et al., “3D Shape Nets: A Deep Representation for Volumetric Shape Modeling, ” Proc. of IEEE Conf. on CVPR, 2015.