

3次元モデルデータの形状特徴抽出に関する研究

馬庭 尚志[†] 柴田 滝也[‡]

[†] 東京電機大学大学院情報環境学研究所 〒270-1382 千葉県印西市武西学園台 2-1200

[‡] 東京電機大学情報環境学部 〒270-1382 千葉県印西市武西学園台 2-1200

E-mail: [†] den03643@nifty.com, [‡] shibata@sie.dendai.ac.jp

あらまし 近年、我々の扱うデータは膨大な量となりデータ群の中から如何にして必要なデータを選び出すかということが重要な鍵となっている。その中でも設計の分野を中心に形状の類似度をキーとした3次元モデルデータの検索システムが必要とされている。我々は、人間の視覚情報処理が幾何学的な特徴を用いた従来の特徴抽出手法よりも優れていると考えた。本稿では、Gaborフィルタによる3次元モデルデータの形状特徴の抽出と、抽出した特徴を用いた3次元モデルデータ検索システムを提案する。本研究では、Princeton Shape Benchmark (PSB)を用いてシステムを構築し、First Tier, Second Tier, Nearest Neighborを算出することで検索精度の評価を行った。その結果、本手法の検索精度は従来手法LFDに近い検索精度があることがわかった。

A Research on Extracting Shape Features from 3D Model Data

Naoshi MANIWA[†] Tatsuya SHIBATA[‡]

[†] Graduate School of Information Environment, Tokyo Denki University 2-1200 muzaigakuendai, Inzai-shi, Chiba 270-1382, Japan

[‡] School of Information Environment, Tokyo Denki University 2-1200 muzaigakuendai, Inzai-shi, Chiba 270-1382, Japan

E-mail: [†] den03643@nifty.com, [‡] shibata@sie.dendai.ac.jp

Abstract It is difficult to find the data from a lot of data groups. A retrieval system of 3D model data is needed in the field of the design. We think that a feature extraction by human visual processing is better than the conventional feature extraction by geometrical characteristics. In this paper, we propose a feature extraction method of 3D model data by using a Gabor filter, and construct the system to retrieve 3D model data. We test the Princeton Shape Benchmark (PSB) to evaluate the system. We evaluate the system by calculating First Tier, Second Tier, Nearest Neighbor. As a result, retrieval precisions by our method are as good as LFD.

1. はじめに

近年、我々の扱うデータは膨大な量となり、データ群の中から如何にして必要なデータを選び出すかということが重要な鍵となっている。その中でも設計の分野を中心に、形状の類似度をキーとした3次元モデルデータの検索システムが必要とされている。データの検索方法として、データの内容そのものをキーとして利用するものがある。この検索方法を用いると、ただ1つのデータから大量のデータを得ることができる。しかし、このような手法を用いる場合、キーとなる対象物の特徴を表す特徴量を抽出することが必要となる。

特徴抽出手法として、3次元モデルデータを構成する頂点データの密度分布を用いる手法や[1]、3次元モ

デルデータを構成する任意の2点間のユークリッド距離を用いる手法[2]、3次元モデルデータの見掛けに基づいた特徴抽出[3]、点群表現を用いた多重解像度表現によるアプローチ[4]など、様々な特徴量抽出手法が提案されている。しかしこれらの手法は3次元モデルデータの幾何学的な特徴を使用するものが多く、人間が実際に見た感じ方とは異なっている可能性がある。

本研究では、3次元モデルデータを複数の視点でレンダリングし、得られた画像に対して、人間の初期視覚野のモデルとしてよく知られるGaborフィルタを適用し、フィルタに対する反応の大きさを形状特徴として抽出した。そして、抽出した特徴を使用して検索システムを構築し、検索精度の評価を行った。

2. 複数の視点から見た3次元モデルデータ

本研究では、3次元モデルデータを複数の視点でレンダリングする。レンダリング画像は、各視点からの平行投影によって生成し、物体の大きさは一律に正規化する。画像のピクセル値は物体が在るか無いかの2値によって表現する。3次元モデルデータを複数の視点でレンダリングした様子を図1に示す。

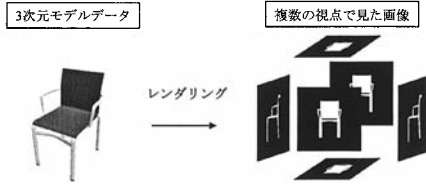


図1 複数視点画像の生成

平行投影によって生成した2値画像は表裏で同じ画像になるため、本手法では表側から見た画像のみを用いることとする(図2)。

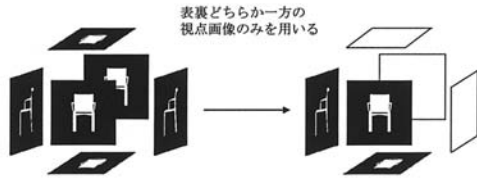


図2 視点画像の選択

3. Gabor フィルタ

Gabor フィルタは、人間の脳の初期視覚野(V1)のモデルとしてよく知られているフィルタリング手法であり、特定の方位成分に反応するという特性を持っている[5]。Gabor フィルタの出力例を図3に示す。

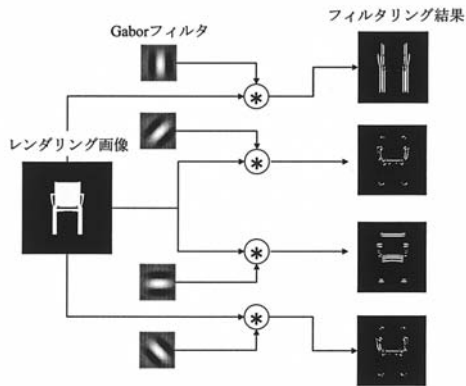


図3 Gabor フィルタ

Gabor フィルタは以下の式で表される。

$$G(i, j) = \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2}\left\{\frac{(i-n/2)^2 + (j-n/2)^2}{\sigma^2}\right\}} \cos(fx\pi(i-n/2) + fy\pi(j-n/2)) \quad (1)$$

$$\left. \begin{aligned} f_x &= kf \cos(a) \\ f_y &= kf \sin(a) \end{aligned} \right\} \quad (2)$$

式中の i, j は Gabor フィルタの座標値, f は周波数, a は関数の傾き, σ は関数の広がり, n はフィルタサイズ, k は任意の定数である(但し σ は通常 n に従属する)。各パラメータを変化させたときの Gabor フィルタの出力を図4に示す。

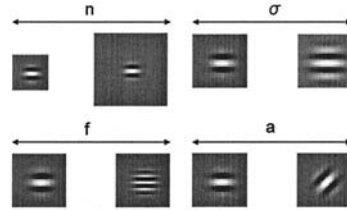


図4 Gabor フィルタの出力例

本研究では Gabor フィルタを3次元モデルデータの各視点画像に適用し、得られた反応の大きさ(図3フィルタリング結果の白色部分の総和)をその方向の形状特徴量として使用する。Gabor フィルタの反応の大きさは、画像中の方位成分の量に比例するから、今回使用する特徴は方位成分の量を抽出する特徴と考えることができる。本手法では、画像中の方位成分の連続性に着目し、パラメータ f, σ, n を固定し、 a だけを変化させたものを一組の特徴ベクトル(GF)として考えた。形状特徴量の算出は、上記の式で表される Gabor フィルタと対象画像との局所的な畳み込みを行うことで行われる。座標 x, y におけるレンダリング画像のピクセル値を $I(x, y)$ とすると、図3のフィルタリング結果の座標 x, y における値 $R(x, y)$ は以下の式によって得られる。

$$R(x, y) = \sum_{i=1}^n \sum_{j=1}^n G(i, j) I(x+i, y+j) \quad (3)$$

画像の横幅と縦幅をそれぞれ、width, height とすると、本手法の i 番目の方位特徴 GF は以下の式で表される。

$$GF(i) = \sum_{y=1}^{height-n} \sum_{x=1}^{width-n} R(x, y)^2 \quad (4)$$

$R(x, y)$ は正と負の両方の値を取るが、今回は正負に関係なく反応の大きさを得るために二乗値を算出する。

4. 相違度の算出

4.1. 特徴ベクトル間の相違度

本研究で使用する特徴は、画像に含まれる方位成分の大きさを抽出するものであり、各方位特徴間で連続性がある。そのため、画像の回転に対してロバストな特徴を構成することができる。例えば、Gabor フィルタの方位特性が $\pi/4$ 刻みのフィルタを構成した場合、図 5 のように、物体を回転しても抽出される特徴の位置が変わるだけで、大きさには変化が無い。

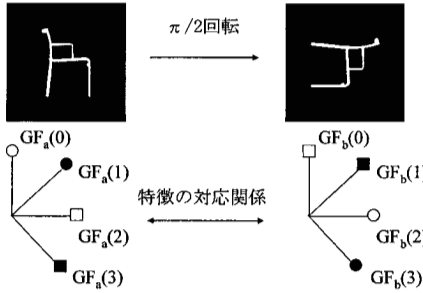


図 5 Gabor フィルタの回転ロバスト性

ここで、2 枚の画像から抽出した特徴ベクトル間のユークリッド距離をその画像間の相違度とし、ユークリッド距離算出の際に、片方の特徴ベクトルの要素を 1 つずつ巡回シフトさせながら 2 ベクトル間のユークリッド距離を求めていくと、求めたユークリッド距離の中で最小のものを 2 枚の画像間の相違度として用いることで、物体が回転していても同一の物体として認識することができる。

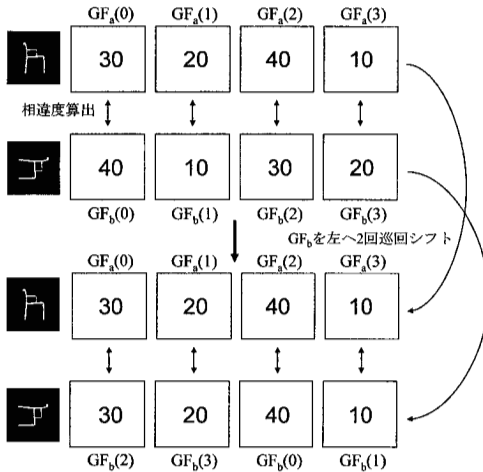


図 6 特徴ベクトル間相違度の算出

例えば、図 5 に示した 4 個の要素を持つ特徴ベクトル GF_a と GF_b の場合では、本手法による両ベクトル間の相違度算出は、図 6 のようになる。特徴ベクトル GF_a と特徴ベクトル GF_b の元となった物体が、画像中の物体の回転を除いて同一のものであったとすると、図 6 のように片方の特徴ベクトルを巡回シフトさせることで特徴ベクトルが一致する。

特徴ベクトルの要素数を m とすると、2 枚の画像から抽出した特徴ベクトル GF_a と GF_b の巡回シフト毎のユークリッド距離 $dist_i$ は以下の式で表される ($i=0, \dots, m-1$)。

$$dist_i = \sum_{j=0}^{m-1} (GF_a(j) - GF_b((j+i) \bmod m))^2 \quad (5)$$

本手法では、このようにして求めた $dist_i$ の中で値が最小となるものを 2 画像間の相違度 D として用いる。

4.2. 3 次元モデルデータ間の相違度の算出

本研究では、3 次元モデルデータを複数視点でレンダリングし、得られた画像に対して特徴抽出を行う。レンダリングされた画像間の相違度は 4.1 節で述べた方法により求めることができるが、3 次元モデルデータ間の相違度の算出にはまだひとつ問題がある。

3 次元モデルデータ A, B を p 個の視点でレンダリングし、特徴を算出した後、全ての視点画像間の相違度を算出すると、 p^2 個の特徴ベクトル間の相違度が算出されることになる(図 7)。

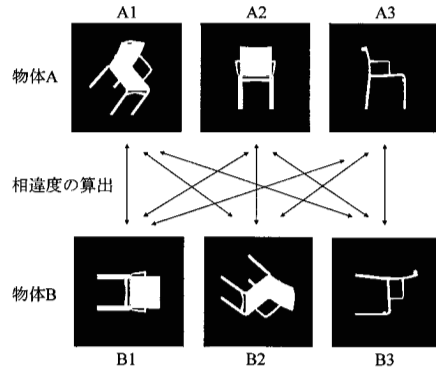


図 7 3 次元モデルデータ間相違度の算出 ($p=3$)

ここで、物体間の相違度を最小化するように、全ての相違度の組み合わせを求めていくと、その組み合わせは p の階乗通りになり、この組み合わせは p が小さい場合には計算可能だが、 $p=100$ など視点の数が多い場合には組み合わせの数が膨大になってしまい、解を求めることができなくなる。本研究ではこの問題に対

して、求められた複数の相違度の中から最小のものを1つずつ選び出すことで、3次元モデルデータ間の相違度を求めた。本手法の流れを図8に示す。

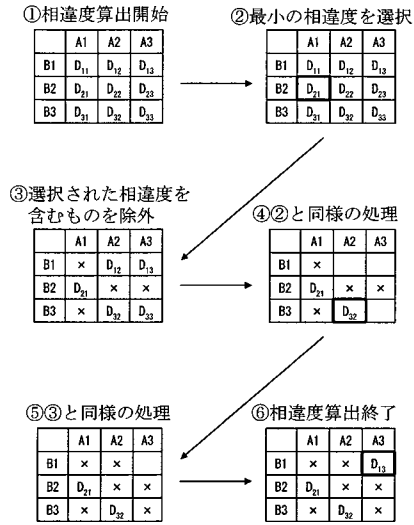


図8 3次元モデルデータ間の相違度算出の流れ

表中の D₁₁, ..., D₃₃ は各視点画像間の相違度である。本手法では、このようにして得られた視点数分の相違度 D の合計和を、最終的な 3次元モデルデータ間の相違度として用いる。

5. 形状類似検索システム

本研究で構築した形状類似検索システムの概略を図9に、実際のシステムのインターフェイスを図10に示す。形状類似検索システムは、ユーザからデータベースに登録されている任意の3次元モデルデータを検索キーとして受け取り、検索キーサンプルの形状特徴量をとり出し、とり出した検索キーサンプルの形状特徴量とデータベース内に登録されている他の3次元モデルデータの形状特徴量との相違度を算出し、相違度の小さい順に検索結果を出力するものである。

6. 評価実験1

6.1. 評価方法

検索精度を評価するために、評価を行った。実験には Princeton Shape Benchmark (PSB) [6] をデータベースとして用いた。PSB は米 Princeton 大学の研究者が公開した 3次元モデルデータのデータベースである。このデータベースは、train データベースと test データベースの2つのデータベースにより構成されている。train

データベースと test データベースはともに 907 個のデータから構成され、それぞれがいくつかのカテゴリ分けが行われている。その中で最も基本的なカテゴリ分けは、base 分類というものであり、本研究でも base 分類を用いて検索精度の評価を行う。

データベースの 3次元モデルデータを、解像度 = 256x256pixel で、x 軸、y 軸方向に、各 $\pi/8$ 度刻みで回転させてレンダリングし、得られた画像に対して、方位特性 = $\pi/8$ 度刻みの Gabor フィルタを適用することで、形状特徴の抽出を行った。また、Gabor フィルタのパラメータとして、11x11, 15x15, 19x19, 23x23pixel の4種類の大きさ(n)と、 $\pi/2, \pi, 3\pi/2, 2\pi$ の4種類の周波数(f)を用いた。

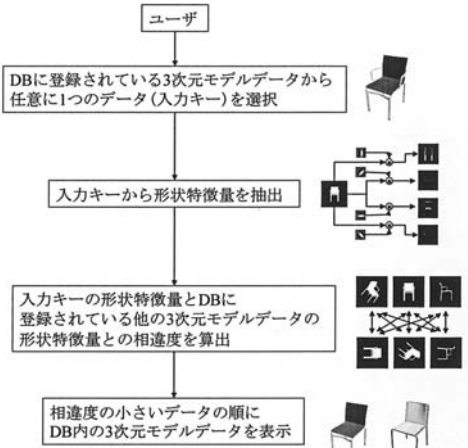


図9 形状類似検索システム

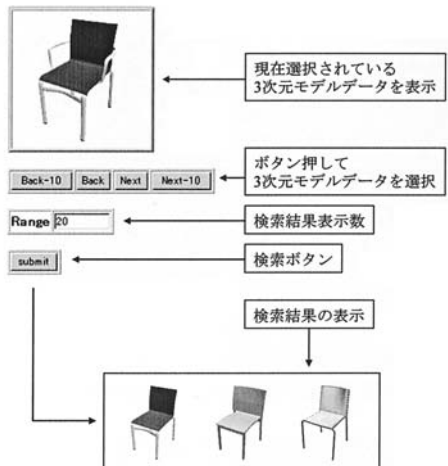


図10 システムのインターフェイス

6.2. 検索精度の評価

検索精度の評価には、First Tier(FT), Second Tier(ST), Nearest Neighbor(NN)を用いた。FTは、カテゴリ c に属するサンプルを検索キーとして検索したとき、検索結果の上位 $q-1$ 個の中に、カテゴリ c のサンプルがどの程度含まれるかを表すものである(q =カテゴリ c に属するサンプルの数)。STは、FTと同様の計算を、上位 $2(q-1)$ 個の場合で求める。NNは、検索をしたとき、検索結果の最上位にくるサンプルが、検索キーと同じカテゴリのものである割合を表すものである。

6.3. 結果および考察

実験によって得られた FT, ST, NN の値をそれぞれ表 1 に示す。

表 1 FT ST NN による検索精度の評価

(n, f, σ)	FT[%]	ST[%]	NN[%]
(11, $\pi/2$, 1)	11.3	17.9	22.8
(11, π , 1)	13.0	20.2	26.5
(11, $3\pi/2$, 1)	14.6	22.0	30.4
(11, 2π , 1)	14.4	22.0	33.4
(15, $\pi/2$, 2)	13.7	20.5	28.6
(15, π , 2)	15.2	22.8	35.6
(15, $3\pi/2$, 2)	25.3	34.8	50.9
(15, 2π , 2)	25.8	36.1	50.1
(19, $\pi/2$, 3)	16.0	23.4	34.1
(19, π , 3)	25.2	34.7	49.0
(19, $3\pi/2$, 3)	25.7	35.9	50.1
(19, 2π , 3)	24.2	34.3	46.4
(23, $\pi/2$, 4)	17.1	23.8	38.3
(23, π , 4)	25.6	36.5	49.3
(23, $3\pi/2$, 4)	24.0	33.8	46.0
(23, 2π , 4)	22.3	32.0	42.4

表 1 を見ると、 $(n, f, \sigma) = (15, 2\pi, 2)$ で全体的に高い精度になっており、フィルタサイズが小さければ周波数の高いもので、フィルタサイズが大きければ周波数の低いもので精度が高くなる傾向がある。今回行ったものと同様の実験が、過去の研究で Shilane らによって報告されている[7]。Shilane らによって報告されている他手法の検索精度を表 2 に示す。

表 2 他手法の FT ST NN の検索精度

特徴量	FT[%]	ST[%]	NN[%]
LFD	38.0	48.7	65.7
REXT	32.7	43.2	60.2
SHD	30.9	41.1	55.6
GEDT	31.3	40.7	60.3
EXT	28.6	37.9	54.9
SECSHEL	26.7	35.0	54.6
VOXEL	26.7	35.3	54.0
SECTORS	24.9	33.4	50.4
CEGI	21.1	28.7	42.0
EGi	19.7	27.7	37.7
D2	15.8	23.5	31.1
SHELLS	11.1	17.3	22.7

表 2 の結果と、本手法の $(n, f, \sigma) = (15, 2\pi, 2)$ のときの結果とを比較してみると、本手法は、FT では SECTORS, ST では SECSHEL, NN では CEGI よりも精度が高いことがわかる。

7. 評価実験 2

7.1. 実験方法

評価実験 1 の結果、本手法は既存の他手法と比較すると精度の不足が感じられた。本手法では、3次元モデルデータから2次元のレンダリング画像を生成し、その画像内の特定方位の線分量を抽出するが、この特徴抽出手法では特徴の詳細度が不足していることが考えられる。例えば、図 10 に示すような2枚のレンダリング画像があった場合、これらの画像から抽出される特徴は同一のものになってしまう。

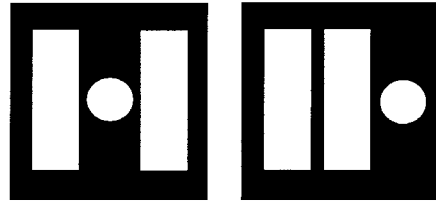


図 10 抽出される特徴が同一のものとなる画像例

この問題は、本手法が画像全体から線分の量を抽出していて、それぞれの線分の位置関係を考慮していないために起こる。本研究ではこの問題を解決するために、レンダリング画像全体から算出していた特徴を、中心から一定の距離区間にわけて算出することで解決した(図 11)。

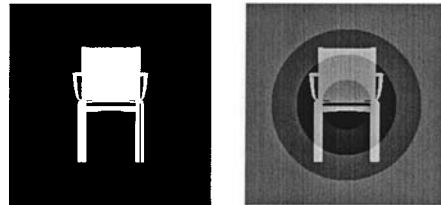


図 11 改良前手法(左)と改良後手法(右)

4.1 節で述べた方法は、一対の特徴ベクトル間の相違度を求めるものであった。今回の特徴の改良によって1枚の画像から複数の距離区間で特徴ベクトルを算出することになったため、本研究では、式(5)で示した相違度算出の方法を拡張した。特徴を s 個の区間にわけて算出したときの拡張された相違度の算出方法は、

以下の式で表される。

$$\begin{aligned}
 dist_i = & \sum_{j=0}^{m-1} \{ (GF_{a_0}(j) - GF_{b_0}((j+i) \bmod m))^2 \\
 & + (GF_{a_1}(j) - GF_{b_1}((j+i) \bmod m))^2 \quad (6) \\
 & + \dots \\
 & + (GF_{a_{s-1}}(j) - GF_{b_{s-1}}((j+i) \bmod m))^2 \}
 \end{aligned}$$

この方法では、複数の特徴ベクトルを同時に巡回シフトさせる毎にユークリッド距離 $dist_i$ を算出させる。このようにして求めた $dist_i$ の中から値が最小のものを相違度 D として用いる。これ以降の相違度算出は、4.2 節で述べた手法に従う。

7.2. 結果

改良手法によって $(n, f, \sigma) = (15, 2\pi, 2)$ で抽出した結果を表 3 に示す。このとき中心からの距離区間をそれぞれ 3, 4, 8, 9 と段階的に変化させた結果、区間を 9 つにわけたときに一番よい精度となった。

表 3 改良手法の FT ST NN

特徴量	FT[%]	ST[%]	NN[%]
分割なし	25.8	36.1	50.1
3分割	31.9	42.2	59.9
4分割	32.8	42.6	62.0
8分割	33.2	42.9	62.2
9分割	33.8	43.3	63.0

表 2 の結果と、改良後の手法によって距離区間を 9 つにわけて $(n, f, \sigma) = (15, 2\pi, 2)$ で特徴抽出をした結果とを比較してみると、改良手法は、FT, ST, NN の全てで LFD に次ぐ精度があることがわかった。特徴算出を複数の距離区間に分けたことで形状特徴の詳細度が増し、その結果、検索精度が向上したと考えられる。

8. おわりに

本研究では、Gabor フィルタによる 3 次元モデルデータの形状特徴抽出に関する手法を提案した。本稿で提案した手法は、複数視点画像と特徴ベクトルの巡回シフトを使用することで、物体の回転に対するロバスト性を確保している。今回の実験で、本手法が過去に提案された手法 LFD に次ぐ精度で検索ができることがわかったが、Gabor フィルタの各パラメータの設定や、4.2 節で述べた 3 次元モデルデータ間の相違度の算出にはまだ改良の余地があると考えられる。本研究の手法は 3 次元モデルデータから複数の視点で見たレンダリング画像を生成し、その画像に対して形状特徴を算出しているため、将来的には、一枚の画像から抽出した形状特徴と 3 次元モデルデータから抽出した形

状特徴を比較することで、各データ間を相互に検索することが可能になると考えられる。今後は、更なる精度向上のために手法の改良を行う予定である。

文 献

- [1] 鈴木一史, 加藤俊一 大津展之: 同値類に基づく回転不変特徴量を用いた 3 次元物体モデルの類似検索, 電子情報通信学会論文誌, Vol.86-D-II, No.8, pp.1234-1243 (2003).
- [2] R.Osada, T.Funkhouser, B.Chazelle, and D.Dobkin: Shape Distributions, ACM TOGS, Vol.21, No.4, pp.807-832 (2002).
- [3] 中澤 昌俊, 武井 毅, 南谷 高宏, 大淵 竜太郎: 見掛けに基づく特徴量を用いた 3 次元形状の形状類似検索, Visual Computing/グラフィックスと CAD 合同ワークショップ 2003, pp.19-24, (2003).
- [4] 武井 毅, 大淵 竜太郎: 多重解像度表現を用いたポリゴンスープ 3 次元モデルの形状類似比較, 画像電子学会誌, 第 34 巻, 第 4 号, pp.256-269, (2005).
- [5] J.D. Daugman: "Complete discrete 2-d Gabor transforms by neural networks for image analysis and compression", IEEE Trans. Acoustics, Speech, and Signal Processing, Vol.36, pp.1169-1179, (1988).
- [6] <http://shape.cs.princeton.edu/benchmark/>
- [7] P.Shilane, P.Min, M.Kazhdan, and T.Funkhouser: The Princeton Shape Benchmark, Shape Modeling International, Genova, Italy.(2004).