

近傍の距離に注目した多様体ランキング法の改良とその3次元モデル検索の応用

栗田侑希紀[†] 大淵竜太郎[†]

本論文では、近傍の距離に注目した多様体ランキング法の改良とその3次元モデル検索への応用を提案する。3次元モデルの形状類似検索における検索精度向上に有効な手法として、特徴空間における特徴点の分布を考慮して類似度計算を行う Manifold Ranking 法がある。しかし、検索要求の小さな近傍（検索結果の最上位付近）に注目すると、単純な距離計算（例えば L2 距離）に比べ Manifold Ranking 法の方が検索精度が低下する場合がある。本研究では、Manifold Ranking を基としながら、単純な距離で得られた検索結果の上位を基にして検索要求を拡張して再度順位付けを行うことにより、順位付けの結果を改選することを目指す。実験的評価の結果、提案手法が検索性能を改善することが分かった。

Improvement of Manifold Ranking Using Neighboring Distance And Its Application to 3D Model Retrieval

Yukinori Kurita[†] and Ryutarou Ohbuchi[†]

This paper proposes an improvement to Manifold Ranking-based ranking of retrieval results in 3D model retrieval setting. Manifold ranking significantly improves ranking results by learning feature distribution in high-dimensional feature space. However, in small neighborhood of the query, ranks obtained by simple distance metric (e.g. that of L2-norm) performs better than those obtained by manifold ranking. Proposed re-ranking method exploits this by using query expansion to improve the ranking result. Experimental evaluation have showed that the proposed method is effective in improving retrieval accuracy.

1. はじめに

近年のコンピュータ記述の発展により、機械設計や医療、ゲームや映画などのさまざまな場面で3次元モデルが利用されるようになってきた。それに伴い、3次元モデルを形状に基づいて比較、検索、分類するための研究がおこなわれている。3次元モデルの形状類似検索システムの大まかな流れを図1に示す。3次元モデルの形状類似検索システムはユーザが検索要求を与えると、要求にそったモデルをデータベース内から検索し、検索結果としてユーザに提示するものである。

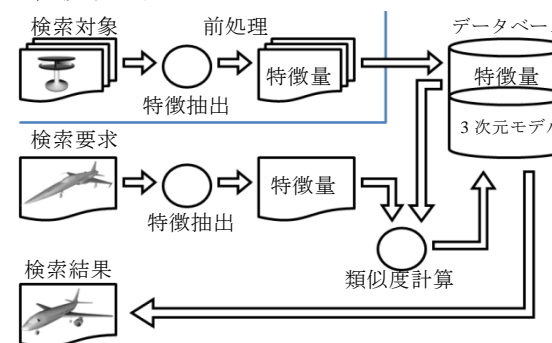


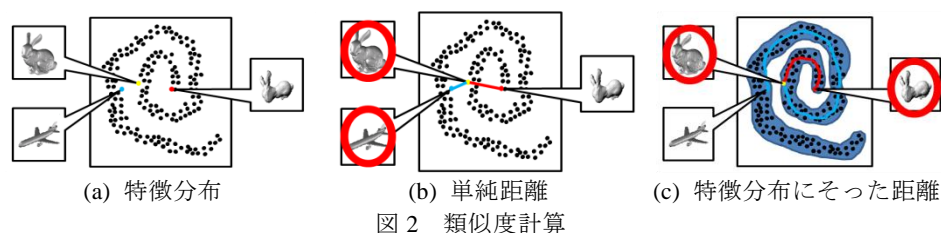
図1 3次元モデルの形状類似検索の流れ

3次元モデルの形状類似検索では、前処理として検索対象となる3次元モデルから特徴量を抽出する。3次元モデルの特徴量は、3次元モデルの形状を数値化した高次元のベクトルであり、検索する際には3次元モデルを直接比較するのではなく、特徴量を用いて比較を行う。検索対象となるすべての3次元モデルについて特徴抽出を行い、特徴量とそれに対応する3次元モデルをデータベース内に保存しておく。以上が前処理となる。次に、実際に検索を行う際の処理を説明する。まず、ユーザが検索要求として3次元モデルを提示する。システムはユーザから提示された3次元モデルに対して検索要求に用いた手法と同じ手法を用いて特徴量を抽出する。そして、抽出した検索要求のモデルの特徴量と、データベース内の検索対象の特徴量について類似度を計算する。特徴量は高次元ベクトルであるので比較する2つの特徴間の距離計算により類似度を計算する方法が一般的である。この類似度計算をデータベース内のすべての3次元モデルの特徴量に対して行い、検索要求に対して、データベース内のどのモデルがより類似しているか、順位づけを行う。最後に順位付けされた結果に基づき、最

[†] 山梨大学
University of Yamanashi

も似ている順に検索結果としてユーザに提示する。

3次元モデルの形状類似検索における検索性能向上の手段の一つとして、3次元モデルの形状特徴量を比較する際に使用する類似度計算の改善が挙げられる。類似度計算の改善手法に **Manifold Ranking**[1]を用いて特徴の分布を考慮し距離計算を行う手法[2]がある。これらの手法は図2(a)のように特徴が分布しているときに、図2(b)のように単純な距離（図では **Euclid** 距離、本論文では特徴の分布に従わない距離ことを単純距離と呼ぶ）で比較するのではなく、図2(c)のように特徴の分布にそって距離を計算する。先行研究では、この手法により検索性能の向上に成功している。



しかし、再現率で **Manifold Ranking** と単純距離を比較すると、検索上位、つまり、検索要求の近傍においては **Manifold Ranking** で得られる特徴分布にそった距離よりも、単純距離のほうがよい順位を与える場合が多いことが分かった。図3はその例である。緑の点線は理想的な検索結果で、検索結果上位から採用する数を増やすに従い、再現率が線形に向上し、正解カテゴリの大きさに達したところで 1.0 に飽和する。これに対し、赤の実線が **Manifold Ranking** による検索結果を、青の破線は単純距離（図では **L0.5** 距離）による検索結果を示す。注目する窓を大きくする、つまり、多数の検索結果を見ると、全体としては **Manifold Ranking** の性能が高い。しかし、詳細に見ると、検索結果の上位（およそ 50 位まで）では単純距離で得られた検索順位が、それ以外では **Manifold Ranking** により得られた検索順位が、相対的に高い再現率を示すことが分かる。

そこで本論文では、検索上位では **L0.5** などの単純距離による順位を、それ以外では **Manifold Ranking** による順位を、それぞれ採用するような順位づけの手法を提案し、これを実験的に評価する。

より具体的には、単純距離によって得られる検索上位の小数を新たな検索要求とし、検索要求集合の拡張 (**pseudo query expansion**) を行い、これを用いて再ランキングを行う手法を提案する。実験的評価の結果、提案する改良が検索性能向上に一定の効果があることがわかった。

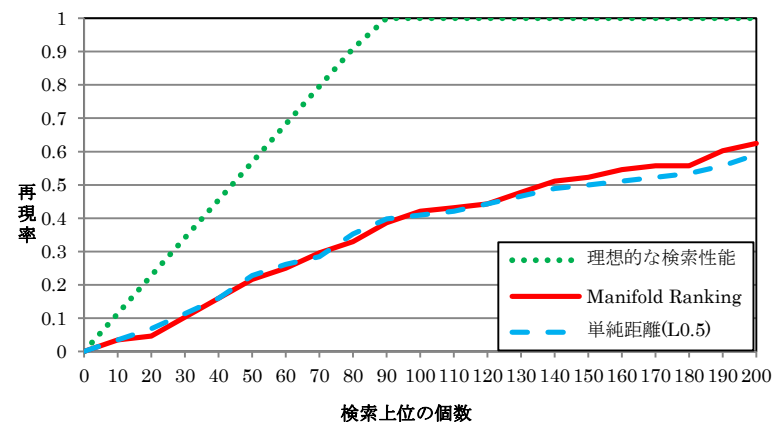


図3 **Manifold Ranking** と単純距離の比較

2. 提案手法

本論文では以下の2点を提案する。

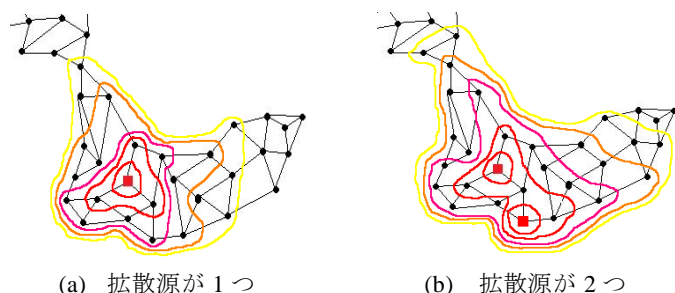
- **Manifold Ranking** を行う際に張るメッシュの削減
- 単純距離と **Manifold Ranking** によるランキング結果を用いた検索要求の拡大

以下にそれぞれの詳細を示す。

2.1 **Manifold Ranking** による類似度計算

提案手法では、Zhou らが提案した **Manifold Ranking** を用いて特徴分布を学習し、特徴分布上での距離計算を行っている。**Manifold Ranking** は、直感的には特徴点を互いの近傍で接続して作成した多様体メッシュに、検索要求の特徴点を源にして有限のインクを拡散してしみこませることにとえられる。拡散した結果、インクの色が濃い部分ほど検索要求に近いことになる。

Manifold Ranking では複数の拡散源を置くこともできることを利用し、有意度フィードバックのフレームワークを使うこともできる[3]。まず、1回目は検索要求1つで検索して結果を得る。次に1回目の検索結果で得られた中から複数の正解モデルをユーザがクリックしてフィードバックする。2回目の検索では、これら複数の正解モデルを拡散源として **Manifold Ranking** を行い、検索結果を順位付ける。こうすると、フィードバックにより検索結果をよりよいものにすることができる。



(a) 拡散源が1つ (b) 拡散源が2つ

図4 Manifold Ranking の模式図

特徴空間 \mathbb{R}^m の点群 $X = \{x_1, \dots, x_q, x_{q+1}, \dots, x_n\}$ があるとする. ここで x_1 から x_q までの q 個の点は検索要求として扱われ, 残りの点群が検索対象である. また, n 次元のベクトル $f = [f_1, \dots, f_n]^T$ を定義する. f_i は x_i についてのランキングスコアを示す. さらに n 次元ベクトル $y = [y_1, \dots, y_n]^T$ を定義する. y_i は x_i が検索対象であれば 1, 検索対象でなければ 0 である. このとき, Manifold Ranking によるランキングスコアは以下のアルゴリズムで求める.

(1) affinity matrix (類似度行列) W を作成する.

$$W_{ij} = \begin{cases} \exp\left(-\frac{d(x_i, x_j)^2}{2\sigma}\right) & \text{if } i \neq j \\ 0 & \text{else } 0 \end{cases} \quad (1)$$

ここで, σ はパラメータである. また, $d(x_i, x_j)$ は点 x_i, x_j の距離である. このとき, すべての点をメッシュで結ぶと, 類似度行列が密行列になり, その空間計算量が $O(n^2)$ になると共に, (3) の逆行列の計算における時間計算量が増加する. また, あまり遠くの点どうしが影響を与えあうと順位付けの結果が悪化することがある. そこで一般的には, 指定した少数 k 個のみを結んでメッシュを作る.

(2) ラプラシアン行列 L を以下により作成する.

$$L = D^{\frac{1}{2}}(D - W)D^{\frac{1}{2}} \quad (2)$$

ここで, D は対角行列であり, D_{ii} には W の i 行目の和が入る.

(3) 類似度 f を解く.

$$f = \left(I + \frac{1}{\mu}L\right)^{-1} y \quad (3)$$

ここで, μ はパラメータであり, $\mu > 0$ である. 以上により求められる f が Manifold Ranking による類似度である. なお, f の値が大きいくほど検索対象と類似していることとなる.

Manifold Ranking はよい順位値を与えることが知られているが, その欠点は時間・空間計算量である. Manifold Ranking では, 類似度行列 W を求めるための k 近傍で結んだメッシュの生成に $O(kn^2)$ の時間がかかる. また, 得られた類似度行列も多くメモリを要する. 点数 n が非常に大きくなると類似度行列がメモリにおさまらなくなる. さらに, 逆行列の計算(3)が高価である. L が疎行列ならば, 逆行列は比較的安定かつ効率的に求めることができるが, それでもその計算量は高い.

2.2 単純距離による類似度計算

単純距離による類似度計算に使われる距離尺度には, Aggarwal らが提案した L0.5 距離[4], Manhattan 距離(L1 距離), Euclid 距離(L2 距離), Kullback-Leibler Divergence (KLD)などがある. L0.5 距離と KLD は以下の式により求めることができる.

$$\text{L0.5 距離} \quad d(x_i, x_j) = \left(\sum_{a=1}^n \sqrt{|x_{ia} - x_{ja}|}\right)^2$$

$$\text{KLD} \quad d(x_i, x_j) = \sum_{a=1}^n (x_{ia} - x_{ja}) \ln \frac{x_{ia}}{x_{ij}}$$

これらの距離を用いて検索要求と検索対象の距離を計算し, 距離が近いものから検索要求に類似しているものとしてランク付けを行う. また, これらの距離は Manifold Ranking において接続行列を作成する際にも用いる. 本論文では予備実験において相対的に検索性能が良かった L0.5 と KLD を使用する.

2.3 検索要求の拡張

提案手法では, 単純距離と Manifold Ranking のランキング結果をもとに, 検索要求とするモデルを増加させることによって検索要求を拡張し, Manifold Ranking によって類似度を再計算する. 検索要求を拡張するために, 検索要求に類似していると考えられる 3次元モデルを検索対象の中からいくつか選択する. そして, 選択した 3次元モデルと検索要求のモデルについて対応するベクトル y の値を 1 に変更する. そして変更した y を用いて, 再度(3)の式により f を解き, 新しい類似度とする.

このとき, 問題となるのは検索要求に類似していると考えられるモデルの選択方法である. 信頼度の低いモデルを使用して検索要求を拡張し, Manifold Ranking で類似度を再計算してしまうと検索性能が下がってしまう可能性がある. そこで本手法では, 図5のようにして使用するモデルを選択する. まず, Manifold Ranking と単純距離のそれぞれを使用してランキングを作成する. 次に2つのランキング結果上位 (図では

上位 4 位まで) を比較し, その 2 つのランキング両方に含まれているモデル (図では丸が付いている 3 種類) を検索要求の拡張に使用する.

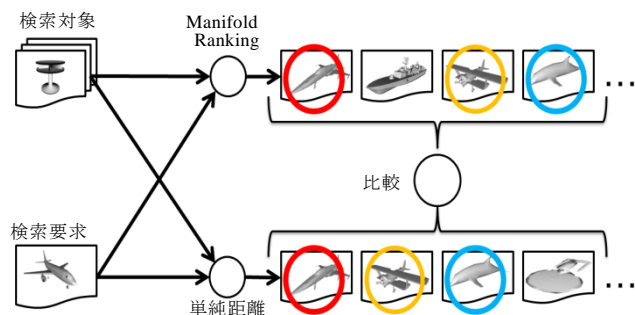


図 5 検索要求拡張に使用するモデルの選択

3. 実験と結果

提案手法の性能を評価するために, 3 次元モデルの形状類似検索を行った. 今回の実験では, モデルの形状特徴として, Wahl らがレンジスキャナで獲得した点群を対象に Wahl らが提案した SPRH[4]をポリゴン等の面からなるモデル向けに改編したものを用いる. SPRH は向き付き点群の点対間の距離と, 点の法線ベクトルの内積等の 3 つの角度からなる 4 次元の変数の結合ヒストグラムで, Wahl らが用いたのは変数ごとにピン数が 5, 全体で 625 次元の特徴ベクトルである. このままではポリゴンの集合やポリゴンメッシュで定義されるモデルには適用できない. そこで, 準乱数を用いてポリゴン面上に面積密度一定で有向点群を生成し, 面ベースの 3 次元モデルから有向点群による 3 次元モデルに変換[6]したうえで SPRH を適用する. 性能評価のために用いるデータセットには SHREC2006[7]を使用した. SHREC2006 は, 人や椅子, 動物, 機械部品などの多様なモデルを含むデータベースであり, 検索対象のモデルが 1, 814 個, 検索対象のモデルを含まない検索要求のモデルが 30 個の計 1,844 個のモデルからなるデータベースである. また, 検索性能の数値的評価として Mean Average Precision (MAP), Mean First Tier (MFT) を使用した. MAP は検索結果に正解のモデルが出現した時の適合率の平均値を全検索要求で平均した値である. MFT はある検索対象のモデルの正解数が n 個としたときに, 検索結果上位 n 個中に含まれている正解モデルの割合である. どちらの指標も値が大きいほど検索性能が良いことを示す.

3.1 Manifold Ranking メッシュの近傍の大きさと検索性能

図 6, 図 7 に Manifold Ranking で用いる近傍の大きさを変化させたときの MAP, MFT

検索性能の変化を示す. グラフの横軸は近傍でメッシュを張る個数, 縦軸は検索性能を示している.

図 6 が示すように, MAP で評価した場合, どちらの距離尺度を用いた場合でも, メッシュを張る個数 k が 50 個以下のときのほうが検索性能が良い. 特に L0.5 ノア倍は明らかな性能のピークが認められる. L0.5 の性能は $k=7$ にピークがあり, $k=25$ 程度まで緩やかに下がった.

図 7 が示すように, MFT で評価した場合, L0.5 は MAP と同じような性能変化をし, $k=7$ にピークがある. しかし, KLD を用いた場合, k が小さすぎると (<7) 性能が悪いが, k が 10 個以上になると性能が頭打ちになり, 検索性能に大きな変化は見られない.

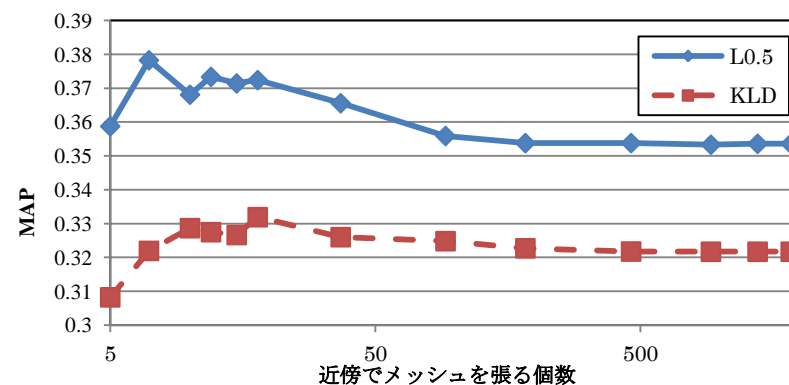


図 6 Manifold Ranking の MAP の変化

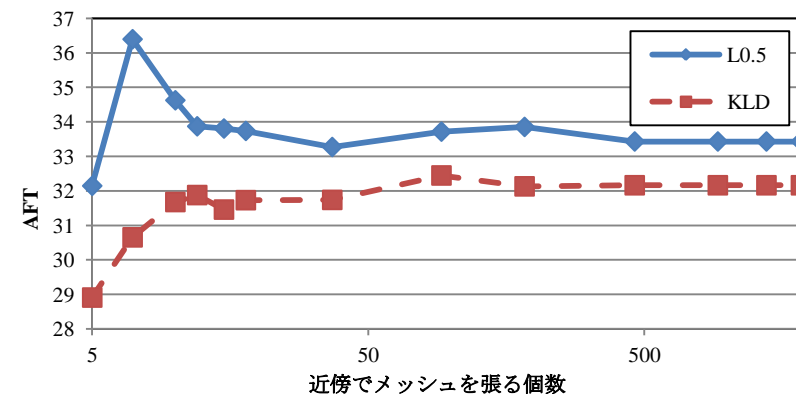


図 7 Manifold Ranking の MFT の変化

3.2 検索要求の拡張

3.2.1 距離尺度 L0.5 を使用した場合

図 8, 図 9 に距離尺度 L0.5 を用いた時, 検索要求を拡張した実験の MAP, MFT を示す. グラフの横軸は検索要求を拡張した個数を表し, 縦軸は検索性能を表している. グラフの左端, 拡張個数 0 は, 提案する検索要求の拡張手法を用いない場合に相当する. 拡張個数 2 とは, 検索要求そのものと, 検索要求のみで得られた 1 度目の検索結果上位 2 個の合計 3 個を拡張された検索要求とし, これら 3 つを拡散源として Manifold Ranking をかけたことを示す.

グラフは, 3.1 節の実験で用いたメッシュを張る近傍サイズ k ごとにプロットしてある. なお, メッシュを張る近傍数が 37 個以上のときについても実験を行ったが, 性能変化の傾向がほぼ同じであったため省略した. 図 8 を見ると, メッシュの近傍サイズ k にかかわらず, 検索要求拡張数が 10 個のときに MAP が最大になることが分かった. また, すべての場合で, 検索要求を拡張しない場合 (つまり拡張個数 0 個) よりも検索性能が向上した. 図 9 は評価尺度 MFT についてプロットしたもので, 多少傾向がつかみにくいですが MAP と同様に拡張個数 10 個で最大の検索性能が得られた.

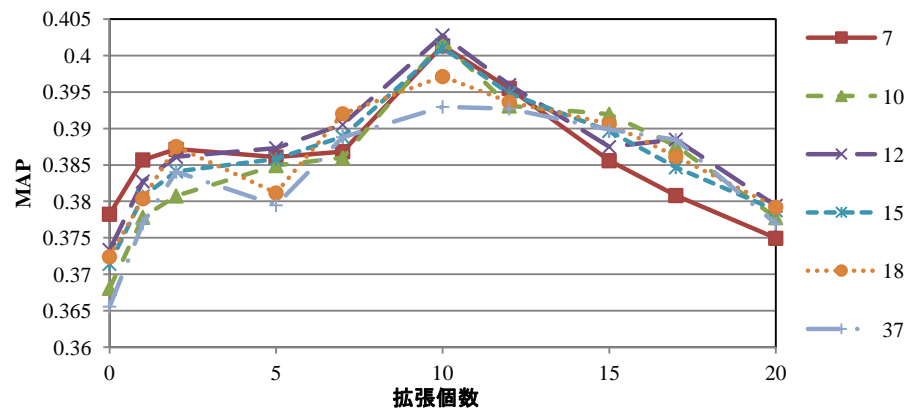


図 8 検索要求を拡張した時の MAP の変化

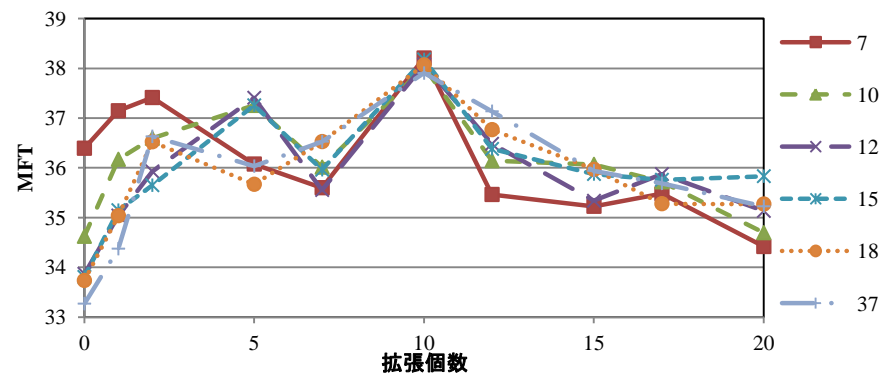


図 9 検索要求を拡張した時の MFT の変化

上記の実験で, MPA と MFT の双方の尺度で最も良い検索性能を得られた拡張個数 10 個の場合について, メッシュを張る近傍 k の個数を変化させたときの検索性能の変化を図 10 と図 11 に示す. これらの図を見ると, 検索要求を拡張した場合には, 近傍サイズ $k=10$ で高い検索性能が得られた. これは提案手法による検索要求を拡張しない 3.1 節の図 6, 図 7 と同じような傾向であるが, ピークの位置は多少ずれている.

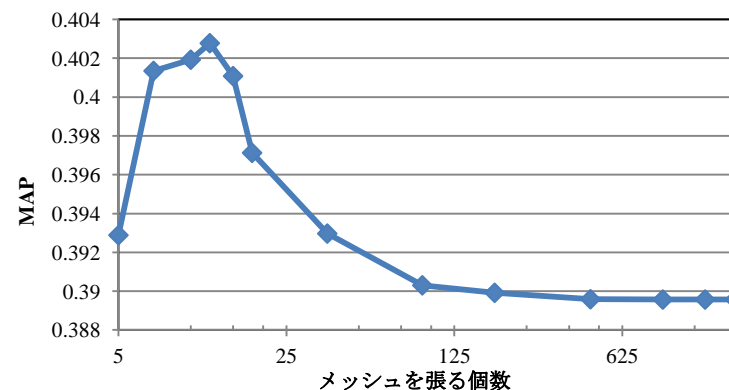


図 10 メッシュを張る個数を変化させたときの MAP の変化

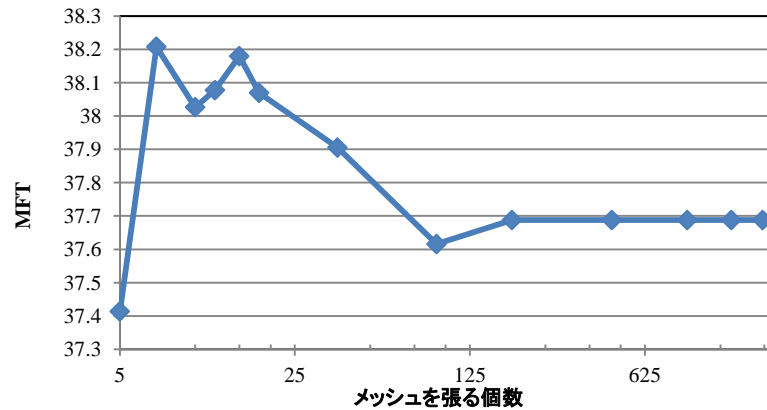


図 11 メッシュを張る個数を変化させたときの MFT の変化

3.2.2 距離尺度 KLD を使用した場合

図 12 と図 13 は、距離尺度 KLD を用いて、検索要求を拡張した場合の拡張個数に対する MAP と MFT の変化を示す。グラフには、3.1 節の実験で用いたメッシュ生成時の近傍個数 k を変えた複数場合についてプロットした。メッシュを張る近傍数が 37 個以上の場合も実験を行ったが、性能変化の傾向がほぼ同じであったため省略した。

KLD の場合も L0.5 距離と同様にすべての近傍サイズ k において提案手法による検索性能向上が見られた。例えば、最も性能が良い拡張個数が 15 個、近傍サイズ $k=10$ から 15 の場合、拡張なしの場合 (拡張個数 0) に比べて MAP が 0.0015 程度向上した。また、図 12 に見られるように、MAP の最大値が 2 か所、拡張個数 2 個付近と拡張個数 15 個付近に現れた。

図 13 のグラフを見ると、MFT で評価した検索性能の最大は、メッシュの近傍サイズ $k=15$ 、拡張個数 2 個の場合の AFT=33.4% である。これは拡張なしの場合 (AFT=31.5%) に比べて 2% 近く向上している。

KLD の場合、MAP、MFT ともに拡張個数 7 個の場合に顕著な谷があるが、その理由はわかっていない。

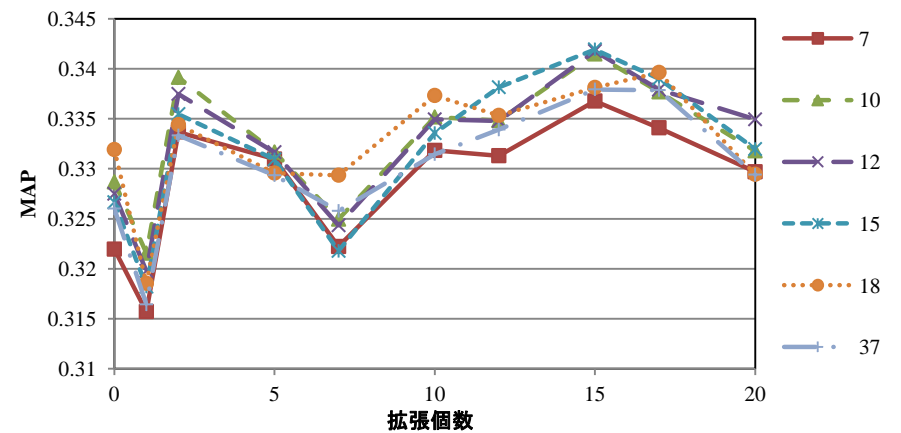


図 12 検索要求を拡張した時の MAP の変化

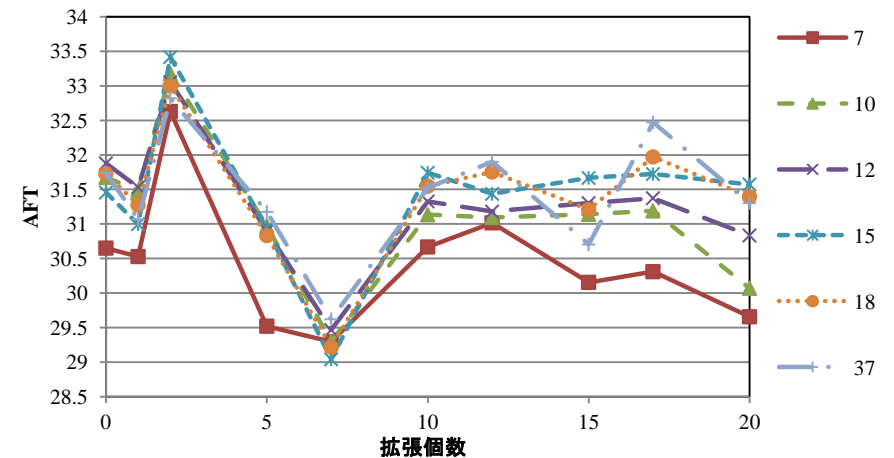


図 13 検索要求を拡張した時の MFT の変化

図 14, 図 15 は検索要求を拡張した実験で MAP, MFT ともに良い検索性能を示した拡張個数 2 個の場合について、メッシュを張る近傍数 k を変化させたときの検索性能をプロットしたものである。実験結果より、検索結果を拡張した場合でも、拡張をしない 3.1 節の図 6, 図 7 と同じような傾向であるが、検索性能が最大となる近傍サイ

ズが移動し、近傍サイズ $k=10$ から 15 付近で性能のピークが見られる。

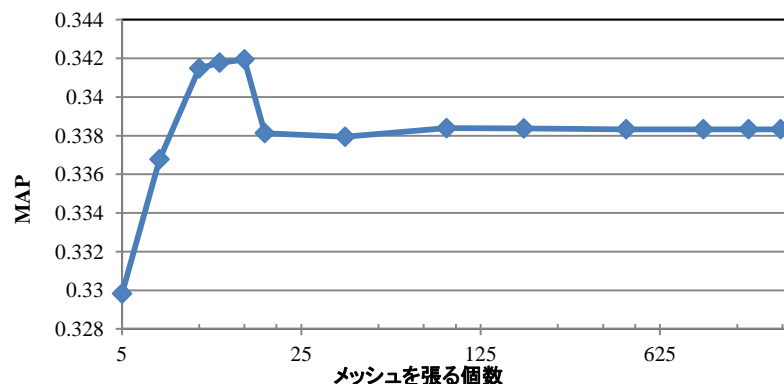


図 14 検索要求を拡張した時の MAP の変化

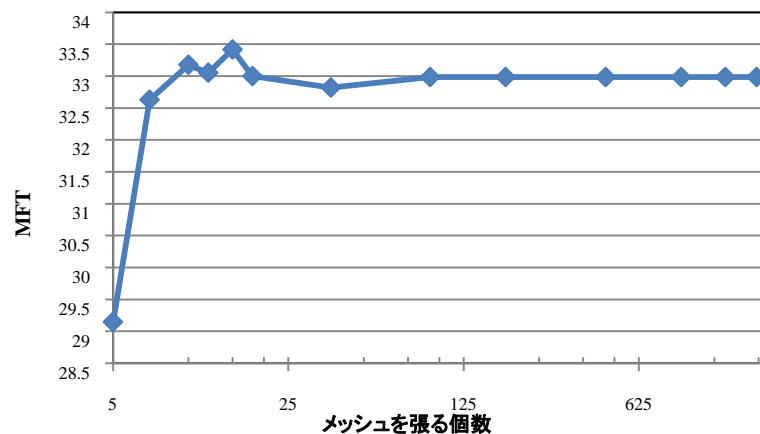


図 15 検索要求を拡張した時の MFT の変化

4. まとめと今後の課題

本論文では、近傍の距離に注目した多様体ランキング法の改良とその 3 次元モデル検索への応用を提案した。本手法では Manifold Ranking の改良手法として、Manifold

Ranking を行う際に使用するメッシュの削減と、Manifold Ranking と単純距離によって得られる検索上位の少数の検索結果を拡張した検索要求とし、これを用いて再ランキングを行う手法を提案した。実験の結果、提案手法による検索の拡張は検索性能向上に寄与することが分かった。使用した距離尺度によって違いがあるが、検索要求の拡張数を変えた場合、単純距離で得られた上位 10 個程度の検索結果を新たに拡張した検索要求として用いた場合に最大の検索性能が得られた。これ以上に拡張数を増加させると検索性能が下がっていくが、これは、より信頼度の低い(初回の)検索結果が含まれるようになるためである。

今後の課題として、検索要求を拡張しての再計算の繰り返しが挙げられる。現在は単純距離で 1 回、その結果に基づいて拡張した検索要求集合で 1 回の 2 段階方式である。拡張を繰り返し行った場合について実験をしてみる価値がある。また、Manifold Ranking 法の最大の欠点はその時間計算量、空間計算量である。特に、(サンプル外の)検索要求を与えるごとにすべての特徴点からメッシュを作成し、そのメッシュ上で適合度の拡散を行う必要があり、計算時間もメモリ消費量も大きい。提案した手法を大規模な 3 次元モデルデータベース、例えば Google 3D Warehouse のように 10^6 個を超える要素を持つデータベースに適用するためには、計算量の問題を解決する必要がある。

謝辞 本研究の一部は、科学研究費補助金 課題番号 20500089 による。

参考文献

- 1) D.Zhou, O.Bousquet, T.N.Lal, J.Weston and B.Scholkopf : Learning with Local and Global Consistency, *Proc.NIPS 2003*, 2003
- 2) 立間淳司, 青野雅樹 : 多様体ランキングを用いた 3 次元物体の形状類似検索, 情報処理学会論文誌 Vol.49 No.10 2008
- 3) Ryutarou Ohbuchi, Toshiya Shimizu, Ranking on Semantic Manifold for Semantic 3D Model Retrieval, *Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval (ACM MIR'08)*, pp.411-418, Oct. 2008, Vancouver, Canada.
- 4) C. C. Aggarwal, A. Hinneburg, and D. A. Keim : On the surprising behavior of distance
- 5) David G. Lowe : Distinctive Image Features from Scale-Invariant Keypoints, *Int'l Journal of metrics in high dimensional space*, In *Proceedings of International Conf. On Database Theory*, (2001) *Computer Vision*, 60(2), (2004)
- 6) Ryutarou Ohbuchi, Takahiro Minamitani, Tsuyoshi Takei, Shape-similarity search of 3D models by using enhanced shape functions, *International Journal of Computer Applications in Technology (IJCAT)*, Vol. 23, No. 3/4/5, pp. 70-85, 2005
- 7) R. C. Veltkamp : SHREC2006 3D Shape Retrieval Contest, Utrecht University Dept. Information and Computing Sciences, Tech. Report UU-CS-2006-030 (ISSN: 0924-3275)