動的オブジェクトの継続モニタリングにおける効率化の提案

加藤　靖之† 北川　博之†† 陳　漢雄†† 古瀬　一隆†††

† 筑波大学大学院コンピュータサイエンス専攻　〒 305–8577 茨城県つくば市天王台 1 丁目 1–1
†† 筑波大学システム情報系　〒 305–8577 茨城県つくば市天王台 1 丁目 1–1
††† 白鴎大学経済学部　〒 323–8586 栃木県小山市駅東通り 2 丁目 2–2
E-mail: †s1920644@u.tsukuba.ac.jp, ††kitagawa@cs.tsukuba.ac.jp, †††chx@cc.tsukuba.ac.jp,
††††furuse@fc.hakuou.ac.jp

キーワード　Spatial Keyword point, Spatial database, Grid index

1 はじめに

近年、多くの人がスマートフォンやタブレットなどのモバイル端末からウェブにアクセスしている。スマートフォンなどのモバイル端末では、どこにいても情報の送受信が可能であり、外を歩いている時でもニュースや動画を見ることや、Twitter や Facebook などの SNS に投稿をすることが可能である。また、GPS 機能を内蔵したモバイルデバイスが普及したことで、ウェブ上にアップロードされる動的データとして、位置情報とキーワード情報を持つオブジェクトが増加している。そのため、人を動的オブジェクトとして捉えたサービスの需要が高まっている。位置情報とキーワード情報を持つ動的なデータの活用を考えるにあたって、以下の例が挙げられる。

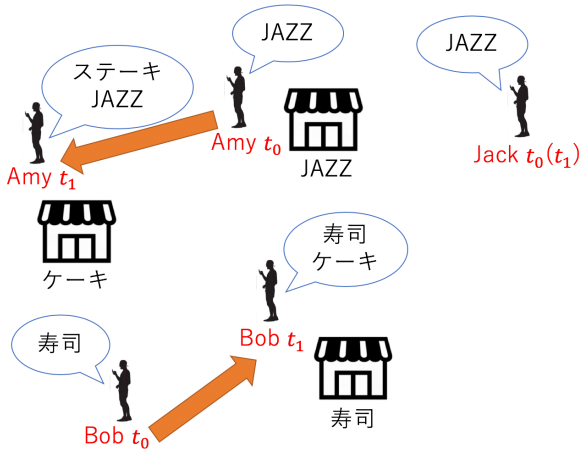

図 1　オンラインクーポン推薦システム

表 1　t_0

SHOP	Top$_1$
JAZZ	Amy
ケーキ	-
寿司	Bob

表 2　t_1

SHOP	Top$_1$
JAZZ	Jack
ケーキ	Bob
寿司	Bob

図 1 はオンラインクーポン推薦システムの例である。寿司屋とケーキ屋と JAZZ 店が存在する地区に、Amy、Jack、Bob の三人が訪れている場合について考える。この例では、それぞれの店がクーポンを送るために最も適切な人を一人が誰であるか監視し続けることを目的としている。時刻 t_0 において、Bob はスマートフォンで近くの寿司屋を検索している。Amy は JAZZ の動画を見ていて、Jack は JAZZ についてのニュースを見ている。このシステムにおいて、Bob は寿司に興味を持ち、寿司屋に近いため、寿司屋の Top-1 となる。また、Amy は Jack よりも位置的に JAZZ 店に近いため、JAZZ 店の Top-1 は Amy となる。誰もケーキについて興味を持っていないため、ケーキ屋の Top-1 はいないものとする。その後、時刻 t_1 において、Bob がケーキについて調べ始めた。その時 Bob の興味のあるキーワードは「寿司」と「ケーキ」の二つとなった。他により位置的に及びキーワード的に寿司に興味を持つ人がいないため、寿司屋は Bob を Top-1 として保持し続ける。そして、ケーキ屋がケーキをキーワードとする Bob を Top-1 のリストに追加した。また、Amy は少し動き、ステーキ店を探し始めた。Jack はその場にとどまり、Amy より JAZZ 店に近くなったため、JAZZ 店の Top-1 が Amy から JACK に変更される。

ここで、Top-K とは、あるデータセットにおいて上位 K 個のスコアを持つデータを取得することを意味する。例の中で使われる Top-1 を考える場合、データを評価したいスコアを用いて降順に並べ、上位 1 件のデータを出力することがこれに当たる。

2 関連研究

関連研究として、キーワード空間データにおいて、オブジェクトのキーワードおよび位置が動的な場合を考える [1] が挙げられる。

キーワード空間データとは、キーワード情報と位置情報を持つデータを意味し、オブジェクトとクエリの 2 種類が存在する。それぞれ位置情報として二次元座標を、キーワード情報としてキーワード集合を持っている。

この研究において、オブジェクトが移動する時、クエリが自身と位置的、キーワード的に最も近い Top-K のオブジェクトを監視し続けることを目標としている。そのため、オブジェクト

が移動する時、クエリは Top-K を維持するために再計算を行う必要がある。しかし、全てのクエリがオブジェクトとのスコアの再計算を行うのではなく、地理的に離れていてオブジェクトの移動が自身の Top-K に影響を与えないようなクエリは再計算を行わない。それにより、再計算の対象を絞るフィルタリングを行なっている。

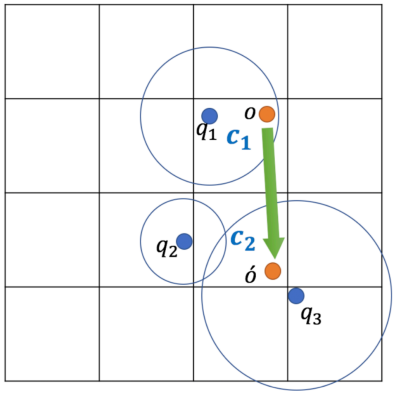

図 2　フィルタリングの例

o は動的オブジェクトであり、それぞれ位置情報とキーワード情報を持つ。q はクエリであり、それぞれの q は Top-K となる K 個のオブジェクトを保持する。q を中心とする円は influential circle と呼び、この円の外部に存在するオブジェクトは地理的に離れすぎているため q の Top-K を更新することがない。また、c はグリッドのセルを表しており、それぞれの c は自身のセルに influential circle が被っている場合、その中心の q を保持する。図の例では、c_1 は q_i のみを保持し、c_2 は q_2 および q_3 を保持する。また、オブジェクト o は自身が Top-K に入っている q を保持しており、図の o は q_1 を保持しているものとする。ここで、オブジェクト o が o' に移動した場合を考える。o' は c_2 に位置しており、セル c_2 が保持する情報から、o' が Top-K に入りうるクエリは q_2 と q_3 のみであることがわかる。また、o を Top-K に含む q_1 に対しても、o' が q_1 の Top-K から抜けている可能性があるため、o' と q_1 の類似度を計算する必要がある。これらから、o が o' に移動する際、o が保持する q_1、c_2 が保持する q_2、q_3 に対してのみ位置情報とキーワード情報をもとに類似度を計算すれば良い。

類似度計算の式は以下のように定義される。

$$SimST = q.\alpha \cdot SimS(o.\rho, q.\rho) + (1 - q.\alpha) \cdot SimT(o.\psi, q.\psi)$$

オブジェクトとクエリの間の類似度は SimST と表す。SimST は位置類似度 SimS とキーワード類似度 SimT の合計の類似度を表している。ここで ρ は位置情報を、ψ はキーワード情報を表す。

SimS はユークリッド距離の類似度であり、データ領域の最大距離である maxDist を使用して

$$SimS(o.\rho, q.\rho) = 1 - \frac{Euclidean(o.\rho, q.\rho)}{maxDist}$$

と表せる。

また、SimT はキーワード $o.\psi$ と $q.\psi$ の TF-IDF を用いて Cos 類似度推定法により求められる。

$$SimT(o.\psi, q.\psi) = \sum_{w \in o.\psi \cap q.\psi} wt(o.w) \cdot wt(q.w)$$

wt(w) はキーワード w の TF-IDF の重みベクトルを表しており、オブジェクトとクエリの重みベクトルは正規化されている。

現状の課題として、位置情報の二次元のみをインデックスとして使用し、絞り込みにおいてキーワード情報を活用できていない点が挙げられる。しかし、キーワード情報を絞り込みに使うためには、空間データとキーワードデータが異なる尺度であるため、活用が困難という新たな問題が発生する。そのため、提案手法では位置情報とキーワード情報の尺度を合わせ、位置情報の二次元のみでなく、キーワード情報を含めた全次元が絞り込みに参加できるような仕組みを提案する。

3　提 案 手 法

ランキング等価を利用し、キーワード情報と位置情報を同一空間上に配置する新しい評価式の提案を行う。これにより、ランキングにおいて空間とキーワードの評価指標の統一が可能となる。また、既存研究の絞り込みは二次元のみと低次元だが、この手法では位置情報の二次元にさらにキーワード次元も追加され高次元となるため、グリッドインデックスを用いた効率的なフィルタリングを提案する。

3.1　ランキング等価

ランキング等価とは、複数のオブジェクトを異なる二つの評価式においてスコア計算をした場合、それぞれの評価式において作成されるオブジェクトのランキングが二つの式で変化しないことを意味する。

以下にランキング等価の定義をあげる。

2 つの距離関数 $dist_1$、$dist_2$ がランキング等価である ⇔ ∀a, b, c, d において $dist_1(a,b) \geqq dist_1(c,d)$ なら $dist_2(a,b) \geqq dist_2(c,d)$

ランキング等価となる式の例として、二点間のユークリッド距離とその二乗をあげる。原点から点 (a, b) および点 (c, d) の距離を考える。距離関数 $dist_1 = \|(x-y)\|$、$dist_2 = \|(x-y)\|^2$ とする。これは明らかに ∀a, b, c, d において $dist_1(a,b) \geqq dist_1(c,d)$ なら $dist_2(a,b) \geqq dist_2(c,d)$ となっている。よって、二点間のユークリッド距離とその二乗はランキング等価と言える。

3.2　ランキング等価条件で類似度計算式の変形

ランキング等価条件において、類似度計算式を変形させる。以後、ランキング等価である式変形を $=_r$ と表記する。

本研究における類似度計算式を以下のように定義する。

$$SimST = q.\alpha \cdot SimS(o.\rho, q.\rho) + (1 - q.\alpha) \cdot SimT(o.\psi, q.\psi)$$

位置類似度 SimS

$$SimS(o.\rho, q.\rho) = 1 - \frac{Euclidean(o.\rho, q.\rho)^2}{maxDist}$$

キーワード類似度 SimT

$$SimT(o.\psi, q.\psi) = cos\theta$$
$$= \frac{|\overrightarrow{o.\psi}|^2 + |\overrightarrow{q.\psi}|^2 - |\overrightarrow{q.\psi} - \overrightarrow{o.\psi}|^2}{2 \cdot |o.\psi||q.\psi|}$$

ここで $|\overrightarrow{q.\psi}| = |\overrightarrow{o.\psi}| = 1$ とすると

$$= \frac{2 - |\overrightarrow{q.\psi} - \overrightarrow{o.\psi}|^2}{2}$$

上記の類似度計算式をランキング等価条件で変形させる。

$$SimST = q.\alpha \cdot SimS(o.\rho, q.\rho) + (1 - q.\alpha) \cdot SimT(o.\psi, q.\psi)$$
$$= q.\alpha \cdot (1 - \frac{|o.\rho - q.\rho|^2}{maxDist}) + (1 - q.\alpha) \cdot \frac{2 - |q.\psi - o.\psi|^2}{2}$$
$$=_r -q.\alpha \cdot \frac{|o.\rho - q.\rho|^2}{maxDist} - (1 - q.\alpha) \cdot \frac{|q.\psi - o.\psi|^2}{2}$$

以上より、SimST は $\sum c_i|x_i - y_i|^2$ の形となり、キーワード情報と位置情報を同じ空間上に配置する評価式を獲得した。

4 クエリ探索

オブジェクト o が o' に移動した場合、全てのクエリに対して類似度を計算すると非効率的である。そのため、類似度を計算するクエリを限定することが必要となる。そのクエリとは、o が Top-K に入っていたクエリ及び o' が Top-K に入りうるクエリの 2 種類である。それら 2 種類のクエリを探索するアルゴリズムを以下に示す。

o が Top-K に入っていたクエリを探索するため、オブジェクト o は自身を Top-K にもつクエリを OutQ として保持し続ける。

また、o' が Top-K に入りうるクエリを探索するため、三角不等式を用いたフィルタリングを行なう。

まず、基準となるクエリを一つ選択する。基準となるクエリは standard query(sq) と表し、o に OutQ が存在する場合、OutQ からランダムに選択し、OutQ が存在しない場合、全クエリからランダムに選択する。$d(a, b)$ を点 a,b 間の距離とした場合、図 3 より、以下の三角不等式が成り立つ。

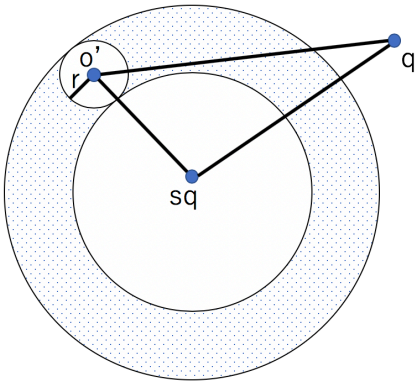

図 3　三角不等式により絞り込み

$|d(sq, q) - d(sq, o')| > r$ ならばクエリ q は o' を Top-K に持たない。

ここで、r とは全てのクエリの中で最大の kScore であり、o' から r 離れたクエリは o' を Top-K に持たないと言える。

証明：三角不等式

$d(sq, o') + d(o', q) \geqq d(sq, q)$より

$d(sq, q) - d(sq, o') > r$は

$d(o', q) > r$

となり、q は o' を Top-K に持たないことがわかる。

クエリは静的オブジェクトのため、$d(sq, q)$ 及び r は既知のものとして、あらかじめ計算する。また、$d(sq, o')$ は一度計算すれば良いため、オブジェクトが移動した時、各クエリに対して類似度を計算する必要がなくなる。

また、sq として複数のクエリを使用することで、フィルタリングをより効率的に行う。

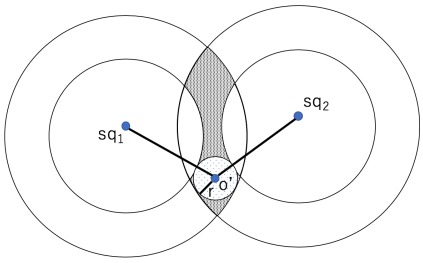

図 4　三角不等式により絞り込み

図 4 に示す通り、sq として複数のクエリを使用した場合、回候補の絞り込みが行なえる。

5 Top-K 保持

動的オブジェクト o が o' に移動した時、Top-K を維持する必要がある。クエリが保持する Top-K に o' が挿入される場合、問題は起きないが、o' が Top-K から抜けた場合、クエリは Top-K のリストを作り直す必要がある。リストの作り直しにかかる計算コストは非常に大きいため、リストの作り直しを防ぐため、Top-K+N を提案する。

5.1 Extra object set(EOS)

リストの作り直しは Top-K からオブジェクトが抜け、Top-K 内の要素が足りなくなった場合に起こる。そのため、オブジェクトが抜けた場合でも要素が足りなくならないように extra object set(EOS) を定義する。

Top-K からオブジェクトが抜けた場合、K+1 番目にスコアの大きかったオブジェクトが新しくクエリの Top-K に追加される。そのため、K+1 番目のオブジェクトが必ず含まれるようなオブジェクトの集合を EOS とし、オブジェクトが Top-K から抜けた場合に EOS の中から K+1 番目のオブジェクトを探索して Top-K に追加する。

EOS は初期集合として、クエリが初期の Top-K を獲得する際に、あらかじめ K+N 番目のオブジェクトまでを計算し、N 個のオブジェクトを EOS として保存する。

オブジェクト o が o' に移動した場合のパターンを図 5 に

示す。図中の kScore は Top-K の最低スコアを表し、eScore は EOS における最低スコアを表す。

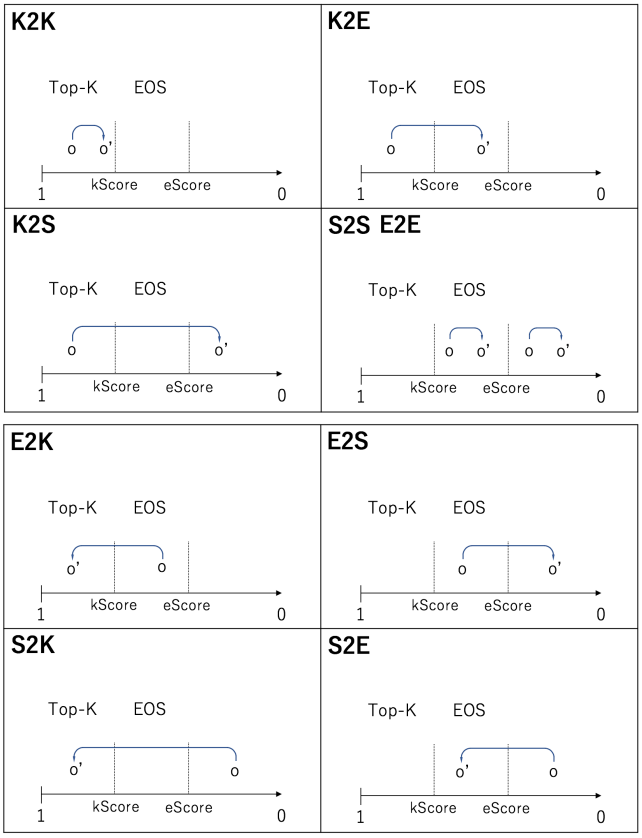

図 5　オブジェクトの移動パターン

K2K. o と o' が両方 Top-K に入っている場合、Top-K の順位を変更する。

K2E. o が Top-K から EOS に移動する場合。o' を EOS に追加し、EOS から k+1 番目のオブジェクトを見つけ、Top-K に追加する。

K2S. o が Top-K から EOS 以下に移動する場合。EOS から k+1 番目のオブジェクトを見つけ、Top-K に追加する。EOS が空の場合、Top-K の作り直しを行う。

S2S,E2E. EOS 内部での移動及び EOS 以下での移動の場合、計算する必要がない。

E2K. o が EOS から Top-K に移動する場合、Top-K をソートし、k+1 番目となったオブジェクトを EOS に追加する。

E2S. o が EOS から EOS 以下に移動した場合、計算をする必要がない。

S2K. o が EOS 以下から Top-K に移動した場合、Top-K をソートし、k+1 番目となったオブジェクトを EOS に追加する。

S2E. o が EOS 以下から EOS に移動した場合、EOS に o' を追加する。

これにより、Top-K の作り直しの回数を防ぐことができた。

6　実　　験

6.1　実 験 環 境

提案手法の有用性を示すために、実験を行った。実験環境をいかに示す。

- OS: macOS 10.13.6
- CPU: 3.1 GHz Intel Core i5
- Memory: 8 GB 2133 MHz LPDDR3
- 開発言語: C++

6.2　データセット

実データを用いて実験を行った。実データは TWITTER からジオタグ付きツイートを取得したデータを用いる。

6.3　パラメータ

出力結果となるクエリサイズ k のデフォルト値は 100、平滑パラメータ α は 0.5、EOS の初期要素数は 5 である。

6.4　実 験 結 果

オブジェクト数、クエリ数、Top-K 数を変更させた実験を行う。提案手法との比較手法として、類似度計算を行うクエリの選定を行っていない base 手法を用いる。

オブジェクト数を変更させた結果を図 6 に示す。オブジェクト数の増加に対する実行時間の減少は base 手法に比べて効率的とは言い難い。

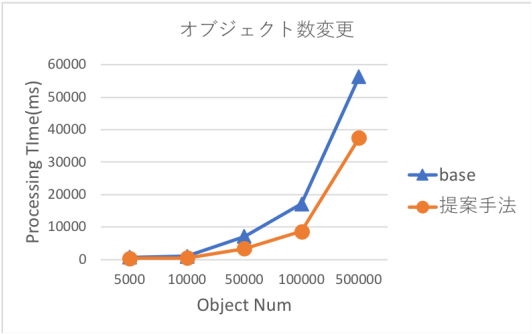

図 6　query=100, k=100

クエリ数を変更させた結果を図 7 に示す。base 手法と比較して、提案手法はクエリ数の変更に伴う実行時間の増加が抑えられていることがわかる。これは、Top-K を更新しないクエリに対して類似度計算を行っていないためであると考えられる。

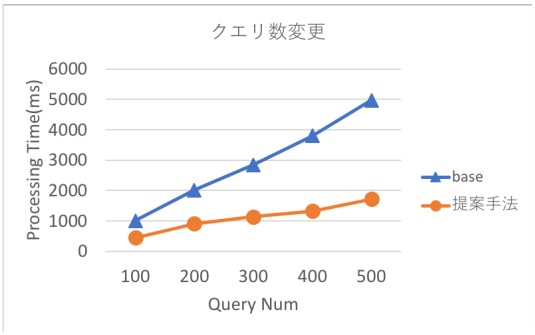

図 7　object=10000, k=100

k の値及び α の値を変更した結果を図 8、図 9 に示す。実験結果から、α の値の変更が実行時間に影響を与えることがない

ことがわかる。これは、位置情報とキーワード情報の評価指標が統一され、同一空間上に配置されているため、それぞれの類似度の重みが同じであるからだと考えられる。

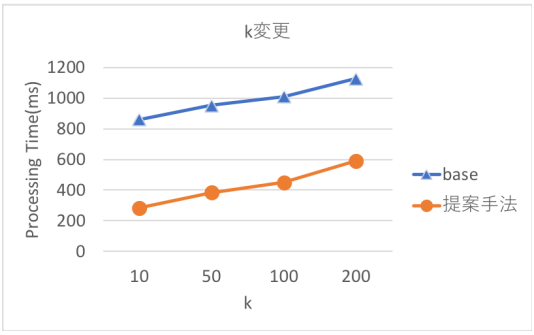

図 8　object=10000, query=100

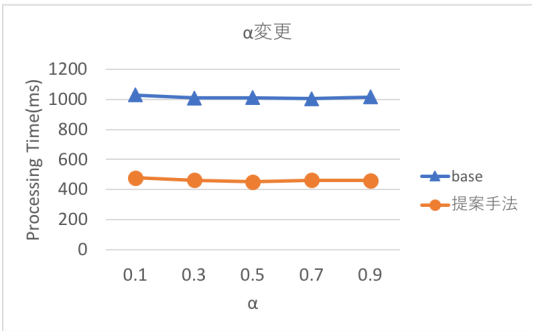

図 9　object=10000, query=100

7　結論と今後の課題

位置情報とキーワード情報を同一空間上に配置し、それら全てをフィルタリングに利用することは、キーワード空間データベースを考えるにあたって重要な問題である。既存研究では位置情報のみを用いてフィルタリングを行っていたが、本研究ではランキング等価を利用し、新たな類似度計算式を定義することで、キーワード情報を用いたフィルタリングを実現することができた。これにより、実データを用いた実験において、クエリ数が増加した場合に効率的なフィルタリングを提案することができたことを示した。

今後の課題として、三角不等式を用いたクエリ探索以外に、vp-tree や Grid などを利用したより効率的なフィルタリング手法を提案することが挙げられる。

謝　　辞

本研究は JSPS 科研費 JP19K12114 の助成を受けたものである.

文　　献

[1] Yuyang Dong, Hanxiong Chen and Hiroyuki Kitagawa, ”Continuous Search on Dynamic Spatial Keyword Objects”, IEEE 35th International Conference on Data Engineering (ICDE), 2019.

[2] Mir Imtiaz Mostafiz, S.M.Farabi Mahmud, Muhammed Mas-ud Hussain, Mohammed Eunus Ali, Goce Trajcevski, ”Class-based Conditional MaxRS Query in Spatial Data Streams”, Proceedings of the 29th International Conference on Scientific and Statistical Database Management, 2017.

[3] Hanxiong Chen, Jianquan Liu, Kazutaka Furuse, Jeffrey Xu Yu, Nobuo Ohbo, ”Indexing expensive functions for efficient multi-dimensional similarity search”, Knowledge and Information Systems, pp165-192, 2011.

[4] L. Chen, G. Cong, X. Cao and K. Tan, ”Temporal spatial keyword top-k publish/subscribe ”, IEEE 31st International Conference on Data Engineering, 2015.

[5] X. Wang, Y. Zhang, W. Zhang, Z. Lin and Z. Huang, ”SKYPE: top-k spatial-keyword publish/subscribe over sliding window ”, Proceedings of the VLDB Endowment, Pages 588-599, 2016.

[6] 獅々堀 正幹, 宮本 裕也, 柘植 覚, 北 研二, ”An improved method to select candidates on metric index VP-tree”, 電子情報通信学会論文誌 91-3, 2008.

[7] 岩崎雅二郎. 木構造型インデックスを用いた近似 k 最近傍グラフによる近傍検索. 情 報処理学会論文誌, Vol. 52, No. 2, pp. 817–828, 2011.