

近似逆最近傍探索による BM25 を用いた映像検索の高速化

的崎 伸彰[†] 岩村 雅一[†] 黄瀬 浩一[†]

[†] 大阪府立大学大学院工学研究科 〒 599-8531 大阪府堺市中央区学園町 1-1

E-mail: matozaki@m.cs.osakafu-u.ac.jp, {masa,kise}@cs.osakafu-u.ac.jp

あらまし 本稿では BM25 を用いた映像検索の高速化を提案する。BM25 は、文書検索に用いられる単語の出現頻度を基にした重み付け手法である。BM25 を使用する際、クエリから抽出した特徴量のデータベース内での出現頻度を求めるために逆最近傍探索問題を解く必要がある。逆最近傍探索問題はデータベースに登録してある各データに最も近いクエリを探す問題であるため、クエリ入力前にできる処理はあまりない。この点において、クエリ入力前にインデクシングすることで大幅な高速化が可能な最近傍探索と様相は大きく異なる。そこで、提案手法では、クエリ側からデータベース側に検索をかけながら、近似的に逆最近傍探索問題を解くことで、映像検索の高速化を目指す。本手法を利用した結果、全探索を用いて出現頻度を求めた場合と比べて、認識率を維持したまま、処理時間を約 3500 分の 1 に削減することができた。

キーワード 映像検索, 近似最近傍探索, BM25, 近似逆最近傍探索

1. はじめに

人には簡単にできるのに、コンピュータには困難なタスクの一つが、映像検索である。人の場合、画像に写っている目標物と同じものが映像に出現しているかを判断することはたやすい。しかし、コンピュータの場合、画像中の目標物と映像中の目標物の向きや照明条件が変化していたり、同じ人物でも服装などが異なると、検索は困難となる。

このような問題への取り組みとして、TRECVID (TREC Video Retrieval Evaluation) [1] がある。これは、映像コンテンツの内容解析及び検索の高精度化を目的とした競争型国際プロジェクトである。TRECVID にはいくつかのタスクが存在し、その中のひとつに Instance Search (INS) がある。INS では、膨大な映像の中から、我々が視聴したい目標物 (物体, 場所, 人物) が写っている画像を使用して映像を検索する。そして、結果は適合度順にランキングして出力する。

このようなタスクに取り組んだ手法として阪田らの手法 [2] [3] がある。この手法では局所特徴量とハッシュに基づく近似最近傍探索 [4] と投票を組み合わせることで高速な検索を実現している。TRECVID 2012 INS において、阪田らの手法は世界第 4 位、日本では 1 位の検索精度を得ている。高速に検索することが可能であるが、検索精度に改善の余地がある。同一のタスクに取り組む、高い検索精度を得ている手法として NTT の手法 [5] がある。この手法は文書検索に使用される重み付けの手法である BM25 [6] を投票に基づく映像検索手法に用いることで高精度な検索を実現している。しかし、BM25 を計算するためには膨大な処理時間が必要となる。

NTT の手法では BM25 を計算するために、クエリから抽出した各特徴量が、データベースに登録されている特徴量の中どの程度出現しているかの頻度を求めている。この出現精度を

求める際、クエリの特徴量を最近傍点とするデータベース中の特徴量を見つける必要がある。これは逆最近傍探索問題 [7] と呼ばれる問題である。逆最近傍探索問題とはクエリが最近傍となるデータベース内の特徴量を見つけるものである。処理時間がかかる原因として、事前のインデクシングの困難さが挙げられる。近似最近傍探索の場合、データベースにあらかじめ前処理をかけることで検索にかかる時間を削減している。処理時間はクエリが与えられてから計測するため、前処理には時間がかかっても問題はない。しかし、逆最近傍探索問題はデータベース側からクエリ側に検索をかけるため、検索される側であるクエリに前処理を行うとその処理も処理時間に含まれてしまう。

そこで、提案手法では近似計算を用いることで、NTT の手法の検索精度を維持したまま、高速化を目指す。このような手法を実現する方法として近似最近傍探索手法の一つである Bucket Distance Hashing (BDH) [8] を用いる方法を提案する。BDH は、多次元ハッシュによって分割された空間とクエリとの推定距離を求めることで、探索範囲を限定する方法である。BDH の処理は距離計算候補の選定と距離計算候補との距離計算の 2 段階に分けられる。提案手法ではある 2 点間の距離はどちら側から見ても同じであるという特性を利用し、探索範囲の絞り込みに BDH を使用することで、近似逆最近傍探索を実現する。これによって、処理時間を大幅に削減できると考えられる。

実験の結果、提案手法は NTT の手法を全探索で実現した場合と比べて、ランキングの上位にどれだけ正解が出現しているのかの指標となる Mean Average Precision (MAP) をほぼ維持したまま、処理時間を約 3500 分の 1 に削減することができた。阪田らの手法と比較しても、処理時間は約 13 倍になったが MAP は約 0.03 向上した。提案手法のパラメータを様々に変えた結果、MAP と処理時間にはトレードオフの関係が見られた。クエリごとの検索結果を見ると、どの手法でも検索に成

表 1 BM25 の各パラメータの対応表

パラメータ	文書検索	映像検索
$Q(= q_1, \dots, q_m)$	文書中の単語群	クエリ画像から抽出した特徴量群
D	データベース中のある文書	データベース中のある映像
$ D $	文書 D の文書長	映像 D から抽出された特徴量数
N	データベースに登録されている全文書数	データベースに登録されている映像数
avgdl	全ての文書の文書長の平均	各映像の特徴量数の平均
$TF(q_i, D)$	文書 D での単語 q_i の出現頻度	映像 D 中に出現する特徴量 q_i の数
$score(D, Q)$	文書 D 中出现する単語 Q によるスコア	映像 D から抽出される Q によるスコア
$n(q_i)$	単語 q_i を含む文書数	同じ映像に登録されている動画の種類の数
$m_{q_i, D}$	文書 D の中で単語 q_i の出現する回数	映像 D の中で特徴量 q_i の出現回数
k_1, b	パラメータ ($k_1 = 2, b = 0.75$)	文書検索と同様

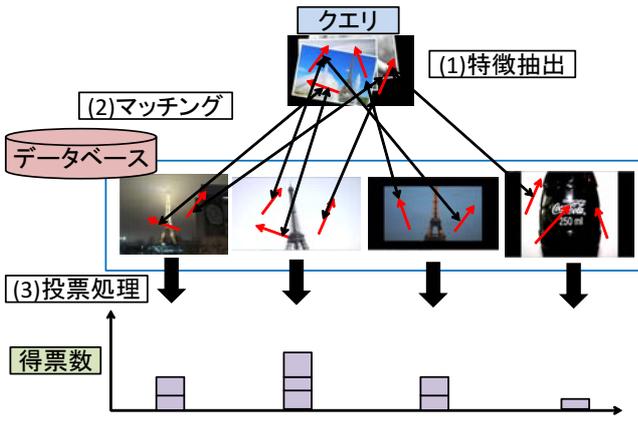


図 1 阪田らの手法の概要

功したクエリと失敗したクエリには同じ傾向が見られた。

2. 映像検索の既存手法

2.1 阪田らの手法

阪田らの手法 [2] [3] では局所特徴量とハッシュに基づく近似最近傍探索 [4] と投票を組み合わせた手法で高速な検索を実現している。阪田らの手法の概要を図 1 に示す。阪田らの手法ではまず、データベース用の映像からフレーム画像を切り出し、局所特徴量を抽出し、データベースに登録する。クエリ画像からも同様に局所特徴量を抽出し、データベースに登録されている特徴量とマッチングをする。そしてマッチした映像に対して投票を行い、得票の多い順に結果を出力している。詳細については以下に述べる。

まず、映像から切り出したフレーム画像から d' 次元の局所特徴量 $x = (x_1, x_2, \dots, x_{d'})$ を抽出する。その特徴量 x の d 次元目 ($d \leq d'$) までを以下の式で 2 値化する。

$$u_j = \begin{cases} 1 & \text{if } x_j - \theta_j \geq 0 \ (0 \leq j \leq d), \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

ここで θ_j は、 x_j の平均値である。これにより、各次元を 2 値化した特徴量 $u = (u_1, u_2, \dots, u_d)$ が得られる。そして、以下の式によってハッシュ値を求め、各次元の値を量子化した特徴量 [4] と映像 ID を組にしてハッシュ表に登録する。

$$H_{\text{index}} = \left(\sum_{i=0}^{d-1} u_i 2^i \right) \bmod H_{\text{size}} \quad (2)$$

ここで、 H_{size} はハッシュ表のサイズである。登録時に衝突が生じる場合、リスト構造によって複数の特徴量を同じハッシュ値に登録する。このとき、ID は異なるが同じハッシュ値に登録される特徴量が多い場合、それらの特徴量は類似しており、認識に十分な識別性能を保持していないと考えられる。そのため、リスト構造によって登録される特徴量に対応する ID の種類に上限を設け、上限を超える場合はそのハッシュ値のリスト全体をハッシュ表から削除し、以降そのハッシュ値への登録を禁止する。以上の処理を全映像から抽出される特徴量に対して行うことにより、データの登録が完了する。

検索の際はデータ登録の際と同様に、クエリから抽出した特徴量 q に対するハッシュ値を計算し、それに基づいてハッシュ表を参照する。そして、得られた特徴量の集合を X とする。次に、 q をスカラー量子化した特徴量と、 X に含まれる特徴量とのユークリッド距離を計算し、上位 k 近傍となる特徴量に対応する ID に投票する。これは、一つのクエリが複数の映像に写っている可能性があるため、一部の正解に票が集まるのを防ぐためである。ここで、データベースの映像毎に抽出される特徴量の数 C_s は異なるため、特徴量の数が多い映像には投票される確率が高くなる。また、投票する k 近傍の特徴量の数が増えてくると、正解以外の映像に投票される確率が高くなる。そこで、第 k 近傍に投票する際に $(0.95)^{k-1} / \sqrt{C_s}$ の重みを付けて投票を行う。クエリのすべての特徴量に対してこの処理を行い、最終的に得票数の多い順に結果を出力する。

探索を行う際、特徴量の各次元の値が撮影条件によって変動する可能性がある。この場合、特徴量から算出されるハッシュ値も変動し、正しく検索ができなくなる。この問題に対処するために、 $q = (q_1, \dots, q_d)$ とするとき、 $|q_j - \theta_j| \leq e$ を満たす次元 j に対しては、 u_j だけではなく $u'_j = 1 - u_j$ も用いて、ハッシュ表を参照する。ここで、 e は各次元の値の変動幅である。この処理を b 個の次元に対して行うことで、 2^b 個のハッシュ値を用いてハッシュ表を参照、探索し、投票する。

2.2 NTT の手法 [5]

本節では NTT の手法について述べる。この手法は、前節で述べた阪田らの手法のように、投票に基づく手法であり、投票

の際に BM25 を組み合わせること高い検索精度を得ている。まず、BM25 について説明し、その後、NTT の手法の流れについて述べる。なお、本稿の実験で用いる手法は、逆最近傍探索を全探索で解いたものであり、[5] に記載の方法とは使用する特徴量が異なる。

2.2.1 BM25

BM25 [6] とは文書 D に対してクエリの集合 $Q = (q_1, \dots, q_n)$ の重みを求める手法で文書検索分野でよく使われている。BM25 は Term Frequency (TF) と Inverse Document Frequency (IDF) を基にした手法である。TF は一つの文書の中で単語 q_i がいくつ出現したかを表している。同じ文書中に何回も出現する単語は、その文書の特徴を強く表現しているという考えに基づいて TF は定義されている。IDF は単語 p が文書を識別する上でどれだけ重要であるかを表している。ある特定の文書にしか出現しない単語はその文書の特徴を強く表現しており、逆に様々な文書に出現する単語は識別性能が低いとみなしている。TF と IDF を基に BM25 は計算されるが、TF と IDF を組み合わせた手法である TF-IDF とは異なり、各文書長を考慮した手法となっている。BM25 のスコアを求める式を以下に示す。

$$\text{score}(D, Q) = \sum_{q_i \in Q} \frac{\text{IDF}(q_i) \cdot \text{TF}(q_i, D) \cdot (k_1 + 1)}{\text{tf}(q_i, D) + k_1(1 - b + b \frac{|D|}{\text{avgdl}})} \quad (3)$$

TF は式 (4)、IDF は式 (5) でそれぞれ表される。

$$\text{TF}(q_i, D) = m_{q_i, D} \quad (4)$$

$$\text{IDF}(q_i) = \log \left(\frac{N - n(q_i) + 0.5}{n(q_i) + 0.5} \right) \quad (5)$$

各パラメータを表 1 に示す。表 1 は文書検索と映像検索において、各パラメータが何を意味しているかを示している。表 1 のように各パラメータを決定することで、文書検索で使用する BM25 を映像検索に対応させている。なお、BM25 の計算時、NTT の手法では IDF に以下の式を使用する。

$$\text{IDF}(q_i) = \log \left(\frac{N - n(q_i)^2 + 0.5}{n(q_i)^2 + 0.5} \right) \quad (6)$$

式 (6) は IDF の識別性を厳しく評価しており、 \log の中身が 1 より小さくなる場合は IDF の値を 0 としている。これによって識別性の低い特徴量を IDF によって枝刈りすることができ、BM25 の性能が向上する。

2.2.2 処理の流れ

まず、データベースの映像のフレーム画像から局所特徴量を抽出する。次に、クエリ画像もクエリと同様に特徴量を抽出し、データベース中の各特徴量がクエリから抽出したどの特徴量にマッチしているかを検索する。マッチングにはコサイン類似度を用い、この値が閾値 th 以上かつ最大となったものとクエリから抽出した特徴量が対応付いたとする。こうしてデータベースの映像から抽出した特徴量全てに対して対応付けを行うことで、BM25 の計算が可能となる。最終的に BM25 による重みを加味した投票を行い、得票の降順で結果を出力する。

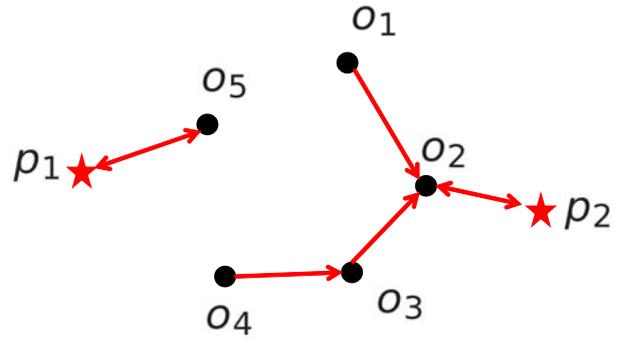


図 2 Monochromatic Reverse Nearest Neighbor (MRNN)

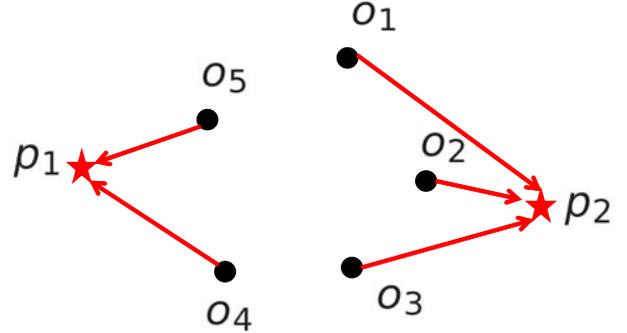


図 3 Bichromatic Reverse Nearest Neighbor (BRNN)

3. 逆最近傍探索

逆最近傍探索 [7] とはデータ集合の中からクエリを最近傍点とするデータをすべて探索する問題である。逆最近傍探索は大きく分けて Monochromatic Reverse Nearest Neighbor (MRNN) と Bichromatic Reverse Nearest Neighbor (BRNN) の 2 つに分類される。

MRNN はクエリとその他のデータを区別することなく、それぞれのデータを最近傍点とする点をすべて求める問題である。具体例を図 2 に示す。 o_1, \dots, o_5 のデータが存在したとき、クエリ p_1, p_2 が与えられたとする。このとき、 p_1, p_2 の逆最近傍探索の解はそれぞれ、 $MRNN(p_1) = \{o_5\}$ 、 $MRNN(p_2) = \{o_2\}$ となる。木構造を用いる方法 [9] [10] や幾何学的アプローチ [11] [12] で解く方法があるが、MRNN は本稿で解くべき問題とは異なる。

BRNN は MRNN とは異なり、クエリとその他のデータを区別し、各クエリを最近傍点とする点をすべて求める問題である。具体例を図 3 に示す。MRNN の例と同様、 o_1, \dots, o_5 のデータが存在したとき、クエリ p_1, p_2 が与えられたとする。このとき、 p_1, p_2 の逆最近傍探索の解はそれぞれ、 $BRNN(p_1) = \{o_4, o_5\}$ 、 $BRNN(p_2) = \{o_1, o_2, o_3\}$ となる。本稿で解く問題は BRNN であり、BRNN を解く手法として [13] や [14] [15] などがある。しかし、BRNN を近似化したものは調査したところ見当たらない。

4. Bucket Distance Hashing (BDH) [8]

一般に近似最近傍探索は大きく分けて 2 段階の処理で行われ

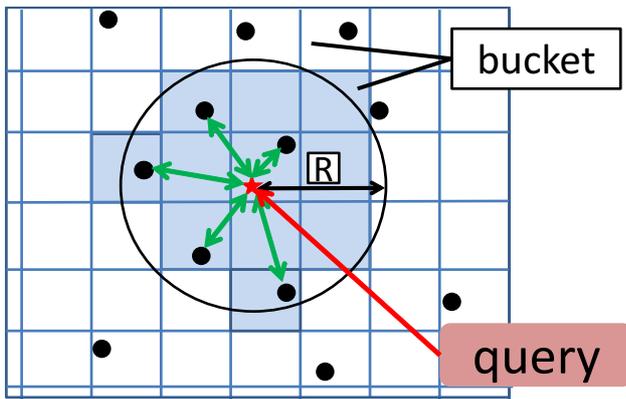


図 4 BDH の概要

る．ひとつは距離計算候補の選定，もう一つは距離計算候補との距離計算である．距離計算候補との距離計算は手法によって大きな差はない．近似最近傍探索毎に異なるのは距離計算候補の選定方法である．代表的な近似最近傍探索手法では，距離計算候補の選定の効率化のために，予め空間をインデクシングしておき，最近傍点と近そうな点が存在する領域のみを選択する．これによって距離計算候補の選定時に各点と距離計算することなく候補の絞り込みを実現している．このような近似最近傍探索手法の一つとして BDH がある．BDH は次元ハッシュによる空間インデクシングによって，分割された空間とクエリとの推定距離を求めることで探索範囲を選択する手法で，現時点で最も高速である．図 4 に BDH の概要を示す．まず，図 4 のように直交基底の直積で空間を分割する．こうして分割された空間それぞれをバケットと呼ぶ．本稿では主成分分析によって得られた上位の主成分基底を直交基底として用いる．

探索時にはクエリと各バケットの重心までの距離を推定距離とし，推定距離がしきい値 R 以下の領域を探索範囲とする．これによって，クエリを中心とする近似的な超球領域に探索範囲を絞り込む．そして，探索範囲内に属するバケットに登録されている特徴量を最近傍候補として類似度を計算する．探索範囲を決定する際，全てのバケットとの推定距離を求めていると，ハッシュ表が大きくなるにつれて推定距離を求めるバケット数は膨大となり，効率的な探索を行うことができない．そこで，経路探索問題に置き換え，効率的な経路を求める枝刈りアルゴリズムを導入することで高速な検索を実現している．

5. 提案手法

提案手法では逆最近傍探索の高速化により，NTT の手法を全探索で実現した場合に比べて，大幅な高速化を目指す．NTT の手法で逆最近傍探索の解を求める際，最近傍探索のようにクエリ側からデータベース側に検索をかけることで解を得ることができれば，最近傍探索に近似を導入して高速化するように，逆最近傍探索も高速化が可能となる．そこで，提案手法では，ある 2 点間の距離はどちら側から見ても同じであるという特性を利用し，候補点の絞り込みにのみ近似最近傍探索を使用する．これによって，出現頻度を近似的に求め，検索精度を維持した



図 5 クエリの例

まま，処理時間の削減を目指す．以下では，処理の流れ，出現頻度の算出について述べる．

5.1 処理の流れ

まず，各映像から一定間隔でフレームを切り出し，それをキーフレームとする．そして，キーフレームから局所特徴量を抽出し，映像 ID と共にデータベースに登録する．検索の際には，データベース作成の際と同様にクエリ画像からも局所特徴量を抽出し，BDH を利用してクエリの特徴量がデータベース内にどの程度出現しているかの出現頻度を求める．出現頻度の求め方については次節で詳しく説明する．こうして得られた出現頻度を基に BM25 を計算し，各映像のスコアを求めて投票する．以上の処理をクエリから抽出したすべての特徴量に対して行い，得票数の多い順に結果を出力する．

5.2 出現頻度の算出

BM25 を計算するにはクエリから抽出した特徴量のデータベース内での出現頻度を求める必要がある．そこで，提案した近似逆最近傍探索手法を使ってクエリから抽出した特徴量 $Q = (q_1, q_2, \dots, q_n)$ (n : クエリから抽出した総特徴点数) と各バケットとの推定距離を計算し，半径 R 内の領域に存在するバケットを探索する．探索の際，BDH では，クエリとデータベース中の特徴量間の距離と相関の高い推定距離を使用しているため，一定の距離内の特徴量の集合を近似的に求めることが可能である．そして，得られた特徴量の集合を基に出現頻度を計算する．このとき，得られた特徴量の集合の中には，実際には逆最近傍探索において正解ではない特徴量も含まれる．そこで NTT の手法と同様に，得られた特徴量とクエリの特徴量とのコサイン類似度を求め，閾値 th 以上の特徴量のみを用いて出現頻度を計算する．

6. 実験

本節では提案手法と NTT の手法（全探索の場合），阪田らの手法の比較を行う．まず実験条件を説明した後，結果と考察を述べる．

6.1 実験条件

データベースとクエリには TRECVID 2012 の Instance Search (INS) のデータセットを用いた．データベースには Flickr の映像が約 200 時間，76751 本用意されている．各映像の再生時間は 10 秒以下である．映像の規模が大きいため，すべてのフレームから特徴抽出を行いデータベースに登録することは現実的ではない．そこで，各映像から秒間 2 フレームを取り

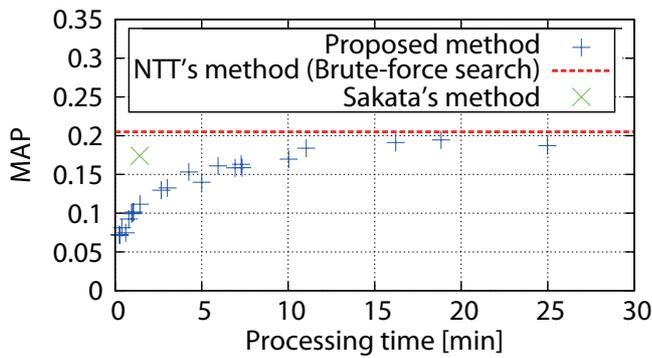


図 6 実験結果

出しキーフレームとする．そして，このキーフレームから特徴抽出を行う．また，クエリは全部で 21 種類あり，各クエリが図 5 のように平均約 5 枚の画像一組として構成されている．検索の際には各組の画像すべてを用いて検索した．特徴点検出には Harris Laplace detector [16] を用いる．この検出器によってスケール変化に頑健な特徴点を得ることができる．そして，特徴量としては色情報を含む特徴量である OpponentSIFT 特徴量 [17] を使用した．ただし，OpponentSIFT 特徴量は 384 次元と高次元のため，主成分分析によって 60 次元に削減して使用した．この 60 次元とは，累積寄与率が約 80% となる次元である．特徴量はすべて長さ 1 に正規化して使用する．評価には上位 1000 件の MAP (Mean Average Precision) を使用する．MAP は複数の正解が存在する場合に用いられる評価法で，ランキングの上位に正解が来るほど高い値となる．INS の正解データは各映像にクエリが出現しているか否かの 2 値で表されており，判断されていない映像も存在する．そのため，評価の際には，そのような映像は除外し，その下の順位の映像を順に繰り上げる処理を行う．コサイン類似度の閾値は $th=0.9$ で実験した．使用した計算機のスペックは Opteron6238 2.60GHz，メモリ 512GB で，計算時間はシングルスレッド換算である．

6.2 実験結果・考察

提案手法において，BDH のパラメータを様々変化させた結果と NTT の手法，阪田らの手法の実験結果を図 6 に示す．NTT の手法では 1 クエリあたり処理時間が約 46 日もかかっていた．提案手法は NTT の手法と比べて MAP をほぼ維持したまま，1 クエリあたりの処理時間を約 3500 分の 1 に削減することができた．また，阪田らの手法と比較すると，処理時間は約 13 倍になったが MAP は約 0.03 向上した．提案手法では探索半径 R を大きくするにつれて MAP は改善した．これは，探索半径を大きくすることで逆最近傍探索において正解である特徴量が探索候補に含まれる数が増加したためである．しかし，探索半径を大きくすると探索候補のバケットが増加し，類似度を求める特徴量の数が増え，結果として処理時間が増加した．この結果，MAP と処理時間にはトレードオフの関係が確認できた．

次にクエリ毎の結果を図 7 に示す．この結果を見ると，どの手法でも AP が高いクエリと低いクエリはほとんど同じであることが分かった．よって，局所特徴量による検索手法では同じような傾向が見られると考えられる．



(a) AP の高かったクエリ例



(b) AP の低かったクエリ例

図 8 クエリ毎の結果

AP の高かったクエリと悪かったクエリ例を図 8 に示す．高い AP を得られたクエリは画面全体に検索対象が映っていたり，同じような背景の映像が正解に含まれていた．そのため，背景が異なるような映像が存在する場合，検索は難しいと考えられる．AP の低かったクエリは検索したい対象物が小さく映っていた．このようなクエリの場合，対象物の領域からは特徴量があまり抽出されず，背景からは多くの特徴量が抽出されるため，背景から抽出された特徴量による誤投票が増え，結果として検索できなかったと考えられる．また，ストーンヘンジの岩や草原のように他の映像でも出現するようなものからは識別性能が高い特徴量が抽出できず，検索することができなかったと考えられる．そして，マクドナルドのロゴは映像によって，色や模様が変化し，認識できなかったと考えられる．そのため，今後これらのようなクエリに対応可能な手法を考える必要がある．

7. ま と め

BM25 を用いた映像検索手法の高速化を実現した．提案手法は NTT の手法を全探索で実現した場合と比べて MAP をほぼ維持したまま，処理時間を約 3500 分の 1 に削減することができた．阪田らの手法と比べても高い MAP を得ることができた．クエリごとの結果を見ると，どの手法でもクエリによって認識できるクエリとできないクエリに大きな差は見られなかった．現状，局所特徴量を用いた手法ではほとんど検索できないクエリも存在するため，特徴点の検出方法など他の方法を考える必要がある．

今後の課題としては，更なる検索精度向上と処理時間の削減を目指して，BDH における推定距離の高精度化や特徴抽出法の改良などが挙げられる．

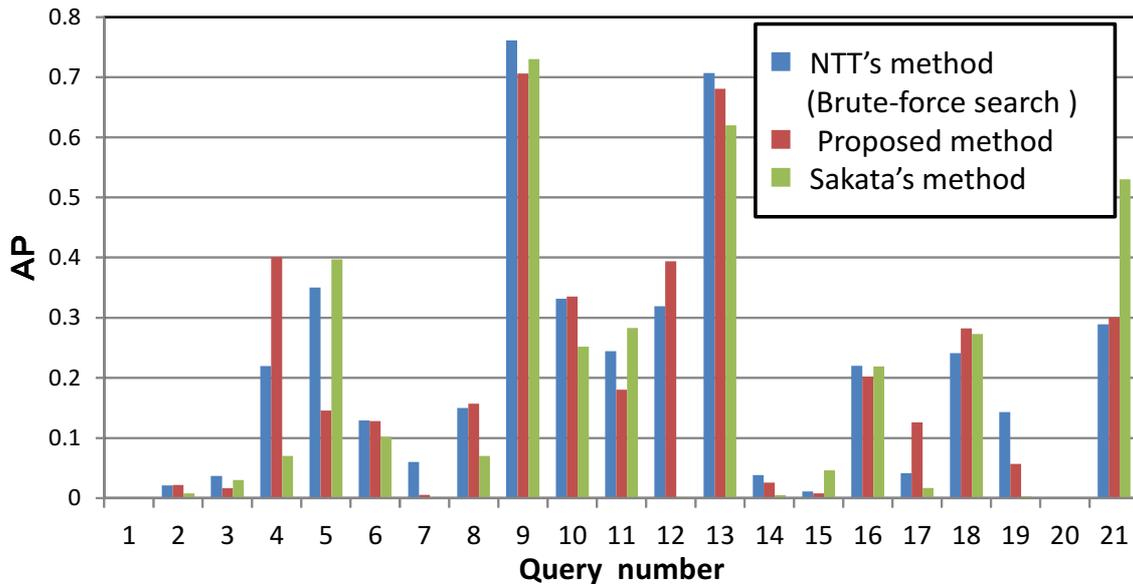


図7 クエリ毎の結果

謝辞

本研究の一部は科学研究費補助金基盤研究 (A)(25240028) の補助による。

文 献

- [1] "TREC video retrieval evaluation". <http://trecvid.nist.gov/>
- [2] 阪田智大, 崎伸彰, 黄瀬浩一, 岩村雅一, "局所特徴量の大規模照合に基づく物体インスタンスの検索 ~ TRECVID2012 Instance Search タスク参加報告 ~", 信学技報, PRMU2012, vol.112, no.441, pp.209-214, Feb. 2013.
- [3] T. Sakata, N. Matozaki, K. Kise, and M. Iwamura, "IMP at TRECVID 2012," TRECVID 2012 Workshop Volume Notebook Paper, pp.163-169, 2012.
- [4] K. Kise, K. Noguchi, and M. Iwamura, "Memory efficient recognition of specific objects with local features," Proc. of the 19th International Conference of Pattern Recognition (ICPR2008), pp.1-4, Dec. 2008.
- [5] 村田真哉, 永野秀尚, 向井 良, 柏野邦夫, 佐藤真一, "画像をクエリとしたインスタンス映像検索", 信学技報, PRMU2012-175, vol.112, no.441, pp.215-219, 2月 2013.
- [6] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: BM25 and beyond," Found. Trends Inf. Retr., vol.3, no.4, pp.333-389, April 2009.
- [7] F. Korn and S. Muthukrishnan, "Influence sets based on reverse nearest neighbor queries," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp.201-212, May 2000.
- [8] M. Iwamura, T. Sato, and K. Kise, "What is the most efficient way to select nearest neighbor candidates for fast approximate nearest neighbor search?," Proc. 14th International Conference on Computer Vision (ICCV 2013), pp.3535-3542, Dec. 2013.
- [9] C. Yang and K.-I. Lin, "An index structure for efficient reverse nearest neighbor queries," Proc. 17th International Conference on Data Engineering, pp.485-492, 2001.
- [10] K.-I. Lin, M. Nolen, and C. Yang, "Applying bulk insertion techniques for dynamic reverse nearest neighbor problems," Proc. Seventh International Database Engineering and Applications Symposium, pp.290-297, 2003.
- [11] I. Stanoi, D. Agrawal, and A. El Abbadi, "Reverse nearest neighbor queries for dynamic databases.," Proc. ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, pp.44-53, 2000.
- [12] A. Singh, H. Ferhatosmanoglu, and A.Ş. Tosun, "High dimensional reverse nearest neighbor queries," Proc. twelfth international conference on Information and knowledge management, pp.91-98, 2003.
- [13] Y. Liu, R.C.-W. Wong, K. Wang, Z. Li, C. Chen, and Z. Chen, "A new approach for maximizing bichromatic reverse nearest neighbor search," Knowledge and information systems, vol.36, no.1, pp.23-58, 2013.
- [14] Q.T. Tran, D. Taniar, and M. Safar, "Bichromatic reverse nearest-neighbor search in mobile systems," IEEE Systems Journal, vol.4, no.2, pp.230-242, 2010.
- [15] J.M. Kang, M.F. Mokbel, S. Shekhar, T. Xia, and D. Zhang, "Continuous evaluation of monochromatic and bichromatic reverse nearest neighbors," Proc. 23rd International Conference on Data Engineering, 2007. ICDE 2007.IEEE, pp.806-815 2007.
- [16] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," Int. J. Comput. Vision, vol.60, pp.63-86, Oct. 2004.
- [17] K.E.A. van deSande, T. Gevers, and C.G.M. Snoek, "Color descriptors for object category recognition," Proc. European Conference on Color in Graphics, Imaging and Vision, pp.378-381, 2008.