

メモリ効率と安定性を向上させた LLAH による カメラベース大規模実時間文書画像検索

竹田 一貴[†] 黄瀬 浩一[†] 岩村 雅一[†]

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

E-mail: [†]takeda@m.cs.osakafu-u.ac.jp, †{kise, masa}@cs.osakafu-u.ac.jp

あらまし 本稿では、大規模文書画像検索を実現するための Locally Likely Arrangement Hashing (LLAH) の改善方法を 3 つを提案する。第 1 は、データベースに登録する特徴点をサンプリングすることによるメモリ削減である。第 2 は、特徴量の次元数を増加させることによる識別性向上である。第 3 は、冗長性を低下させることによる特徴量の安定性向上である。実験により、必要メモリ量が 50%削減され、2000 万ページデータベースにおいて精度 99.3%、処理時間 54.5ms となることが確認された。

キーワード 文書画像検索, リアルタイム処理, LLAH, 大規模データベース

1. はじめに

近年、スマートフォンと呼ばれる携帯電話と携帯情報端末 (PDA) を融合した携帯端末が急速に普及し始めている。これらは、従来の携帯電話とは異なり、本格的なネットワーク機能を備えている。その機能を利用することで、様々な情報検索サービスを提供することが可能である。その 1 つとして、Google が提供する「Google Goggles」[1] がある。「Google Goggles」とは、スマートフォンに搭載されたカメラで撮影した画像を、Google のデータセンターに送信することで、その画像中の物体に関連付けられたデータをユーザに提供するというものである。例えば書籍の表紙を撮影することで、その書籍に関する情報を得ることができる。このようなツールを利用することにより、ユーザがその物体に関する知識を有していなくても、有益な情報を簡単に取得することができる。このようなサービスを実現しうる技術として特定物体認識がある [2]。本稿では特に、その対象を印刷文書に限定した文書画像検索について考える。

文書画像検索とは、印刷文書からその元となった電子文書を検索する処理である。すなわち、印刷文書を文書画像に変換し、文書画像化した電子文書と比較照合する手法である。このような処理を実現するための様々な手法が提案されている [3][4]。しかし、これらは印刷文書を文書画像に変換する入力機器としてスキャナを用いるため、画像取り込みの手間や携帯性などの面から利便性に欠ける。そこで、入力機器としてデジタルカメラを利用することを考える。具体的には、デジタルカメラで撮影した印刷文書の一部あるいは全部を検索質問とし、その検索質問に対応する文書画像をデータベースから検索するというものである。デジタルカメラを利用することで画像入力容易になり、印刷文書を用いた様々なサービスを提供することが可能になる。

デジタルカメラを用いる場合、高い利便性をもつ一方で、撮影画像特有の問題がいくつか生じる。それは、射影歪みや照明変化、ピントのずれなどである [5]。このような様々な外乱にロバストである文書画像検索法として、Locally Likely Arrangement Hashing (LLAH) が提案されている [6]。LLAH では、文書画像から比較的安定して抽出できる単語の重心を特徴点とする。そして、特徴点の配置から計算される特徴量に基づいて検索する。その特徴量として、射影歪みに対して不変となる幾何学的不変量を用いる。また、特徴量は特徴点ごとに計算される。以上の性質から、撮影時に生じる隠れや紙面の湾曲などの外乱にロバストである。また、特徴量の計算量が、検索対象の特徴点数 N に対して $O(N)$ であるため、リアルタイム検索が実現できるほどの高速性を持つ [6]。さらに、多言語で記述された文書画像への拡張もすでに行われている [7]。以上の性質を利用して、カメラペンシステム [8] への適用や、文書への拡張現実 [9] 等の応用がなされている。

しかし、LLAH にはデータベースの大規模化を行う上で 2 つの問題点がある。第 1 に、大量のメモリを必要とすることである。具体的には、2000 万ページの文書画像を登録するために、約 300GB のメモリが必要となる。従って、このようなスケラビリティの悪さを改善しなければ、大規模化は望めない。第 2 に、検索精度が低下するという点である。大規模化に伴い類似した特徴点の配置を持つ文書が登録される可能性が高くなる。その結果、誤投票の増加につながり、検索精度の低下を招く。このような検索精度の低下を防ぐためには、特徴点の配置をより細かく表現しなければならない。つまり、特徴量の識別性を向上させる必要がある。

本稿では、必要メモリ量の削減と特徴量の識別性向上を図るための LLAH の改善方法を 3 つ提案する。第一は、データベースに登録する特徴点をサンプリングすること

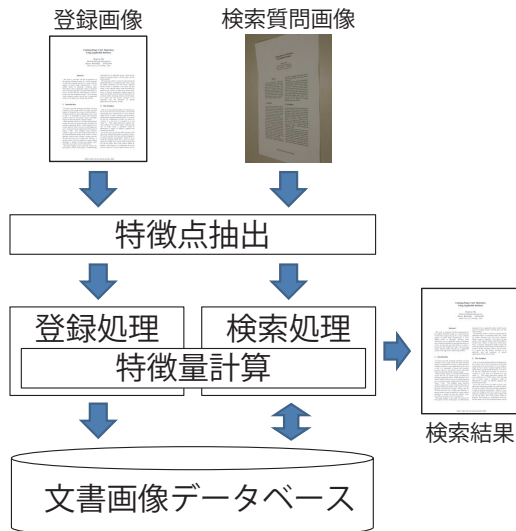


図1 処理の流れ

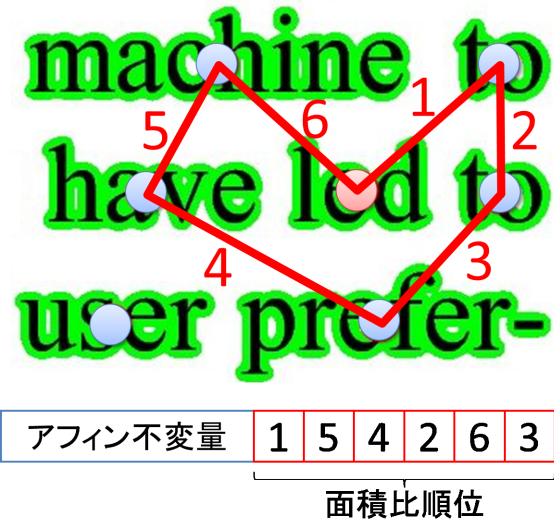


図2 追加特徴量

で、データそのものを削減するメモリ削減法である。第2に、特徴量の次元数を増加させることによる識別性向上である。しかし、次元数を増加に伴い、特徴量の安定性が低下することがわかっている[10]。そこで、特徴量の冗長な次元を削除し、特徴量の安定性向上を図る。実験結果から、約50%のメモリ削減効果を確認した。また2,000万ページデータベースにおいて、検索精度99.3%、処理時間54.5msで検索可能であることを確認した。

2. LLAHを用いた文書画像検索

まず、従来手法であるオリジナルのLLAHについて説明する。

2.1 処理の概要

図1にLLAHを用いた文書画像検索の処理の概要を示す。登録文書画像および検索質問画像は、特徴点抽出処理で特徴点の集合に変換される。その後、登録文書の特徴点は登録処理へ、検索質問の特徴点は検索処理へ入力される。これらの処理では、特徴点の配置に基づく特徴量を計算する。登録処理では、特徴量に基づいて特徴点及び文書の識別番号を文書画像データベースに登録する。つまり、文書画像は特徴点によってインデキシングされる。検索処理では、特徴量を用いて文書画像データベースを参照し、対応点を検索する。対応点を持つ文書の識別番号に投票することで、所望の文書を検索する。以下で各処理について説明する。

2.2 特徴点抽出処理

LLAHでは特徴点の配置に基づいて文書画像のマッチングを行う。従って、特徴点抽出処理では、射影歪みやノイズが生じている場合や、低解像度の場合でも同一の点を抽出する必要がある。そのため、単語領域の重心を特徴点として用いる。

特徴点抽出の手順を以下に示す。まず、入力画像を適

応2値化し、2値画像を得る。次に、2値画像をガウシアンフィルタでぼかし、再度適応2値化を行うと、単語ごとに連結された画像が得られる。最後に連結成分の重心を計算して特徴点とする。

2.3 特徴量計算

特徴量とは、データベースから対応点を検索するためのキーとなるものである。LLAHで用いる特徴量は、ロバスト性や隠れ耐性を実現するため、特徴点ごとに定義される。また、特徴量として幾何学的不変量を用いる。これは、デジタルカメラで撮影された画像において生じる射影歪みに対して不変とするためである。具体的には、局所領域で射影不変量に近似できるアフィン不変量を用いる。アフィン不変量は、同一平面上の4点 $ABCD$ から、以下の式で求められる値である。

$$\frac{P(A, C, D)}{P(A, B, C)} \quad (1)$$

ここで、 $P(A, B, C)$ は頂点 ABC で囲まれる三角形の面積である。この4点で定まる2つの三角形は必ず1辺を共有する。LLAHでは局所領域に存在する特徴点を用いて特徴量を計算するため、射影歪みに対するロバスト性を実現している。

近傍点の配置から、ある特徴点 s の特徴量を計算する場合、最も単純な方法は s の近傍4点から計算されるアフィン不変量を特徴量とするものである。しかし、4点の配置の持つ識別性は高くない。そこで、近傍 $m(> 4)$ 点から4点を選ぶ組合せを考える。 ${}_m C_4$ 通りあるすべての組合せからアフィン不変量を計算し、それらを並べたものを特徴量とする。 m が大きいくほど、同じ m 点が無効な可能性は低くなる。このように計算されたアフィン不変量は、特徴点の位置の変動により誤差が生じる。そこで、特徴量の安定性を高めるため、アフィン不変量を離散化する。最終的に、離散化したアフィン不

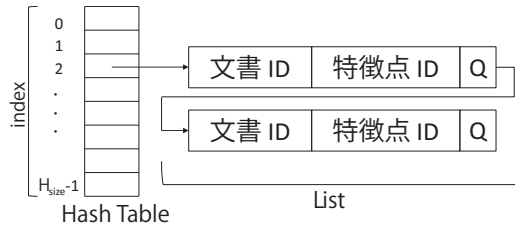


図 3 ハッシュ表の構成

量の列 $(r_{(0)}, r_{(1)}, \dots, r_{(mC_4-1)})$ を特徴量とする。

射影歪みを受けた場合、近傍 m 点として異なるものが得られることがある。一方、近傍 n 点のうち $m (\leq n)$ 点までは、ある程度の射影歪みを受けていても共通のものが得られる可能性が高い [6]。そこで、 n 点から得られるすべての m 点の組合せ ${}_nC_m$ 通りを調べることにする。 n 点のうち、 m 点が共通であれば、同じ m 点の組合せを得ることができる。このようにして特徴量の安定性を実現している。オリジナルの LLAH では、 $n = 7, m = 6$ としている。

特徴量の識別性をより向上させるために、連結成分の面積から得られる特徴量を追加する。まず、 m 点を抽出した連結成分の面積を求める。次に、図 2 に示すように隣り合う連結成分の組に番号を与え、ペアとなる単語領域の面積比を計算する。例えば、1 は “led” と “to” の単語領域の面積比を指し示す番号である。そして、連結成分の組の番号を面積比の大きさの順番で並べる。この数字の列を追加特徴量として、アフィン不変量ベクトルに追加することで、特徴量は $({}_mC_4 + m)$ 次元となる。

2.4 登録処理

登録処理では、特徴量をハッシュ表のインデックスに変換し、それをを用いて文書画像データベースへ特徴点を登録する。ハッシュ表のインデックス H_{index} は以下に示すハッシュ関数で計算される。

$$\left(\sum_{i=0}^{(mC_4+m)} r_{(i)} d^i \right) = Q H_{\text{size}} + H_{\text{index}} \quad (2)$$

ここで d は離散化レベル数、 $r_{(i)}$ は離散化された不変量、 H_{size} はハッシュ表のサイズを表す。このハッシュ関数によって H_{index} と商 Q は、特徴量に対して一意に決定される。そこで、ハッシュ表には図 3 に示すように、文書 ID、特徴点 ID、 Q を 1 組としてを保存する。商 Q を保存することにより、同じ H_{index} を持つ特徴量が一致しているかどうかを、 Q の一致で確認することができる。ここで、文書 ID とは文書の識別番号、点 ID とは特徴点の識別番号である。また、衝突が生じた場合は、図 3 で示すようなリスト形式で追加する。

2.5 検索処理

LLAH では、登録文書への対応点をハッシュから見つけ、その数を投票数として検索を行う。

negative example
applied to fine
criteria for

図 4 サンプリング対象の単語領域

まず、登録処理と同様に各特徴点に対し、ハッシュ表のインデックス H_{index} と商 Q を求める。得られたインデックス H_{index} を用いてハッシュを参照し、図 3 に示されるリストを取得する。リストの各項目について商 Q が一致するかを調べ、一致した場合にそのリストの文書 ID に 1 票投票する。最後に、最大得票数を得た文書を検索結果とする。

3. 提案手法

3.1 メモリ使用量の削減

LLAH では、1 文書あたり約 600 点の特徴点を抽出する。抽出された各特徴点に対して 7 個の特徴量を計算することから、文書 1 ページにつき約 4000 個の特徴量をデータベースに保存することになる。2000 万ページのデータベースを作成する場合、8000 億個もの特徴量を保存することになり、必要メモリ量が膨大になることは明らかである。LLAH は検索に投票処理を用いている。従って、ある程度の特徴点が残っていれば、検索は可能であると考えられる。そこで、ハッシュに保存する特徴点をサンプリングし、メモリ使用量の削減を図る。

特徴点のサンプリング方法の 1 つとして、ランダムサンプリングが考えられる。しかし、この方法では特徴点の位置に偏りができてしまう可能性がある。つまり、文書上での特徴点の分布が疎な領域と密な領域ができるということである。そのような場合、疎な部分を撮影した検索質問では正確に検索できない。撮影する範囲によって検索できない部分が生じることは望ましくないため、サンプリングされる特徴点は文書上に均一に分散していなければならない。

提案手法では、文書全体に分布した特徴点をサンプリングするため、単語領域の面積に着目する。具体的にはまず、周囲の単語領域よりも面積が小さくなるような単語を選択する。そして、その単語から抽出された特徴点をサンプリングする。サンプリング対象となる単語領域の例を図 4 に示す。この例では、“to” という単語の面積が周囲よりも小さいことがわかる。従って、この “to” から抽出された特徴点がサンプリングされることになる。

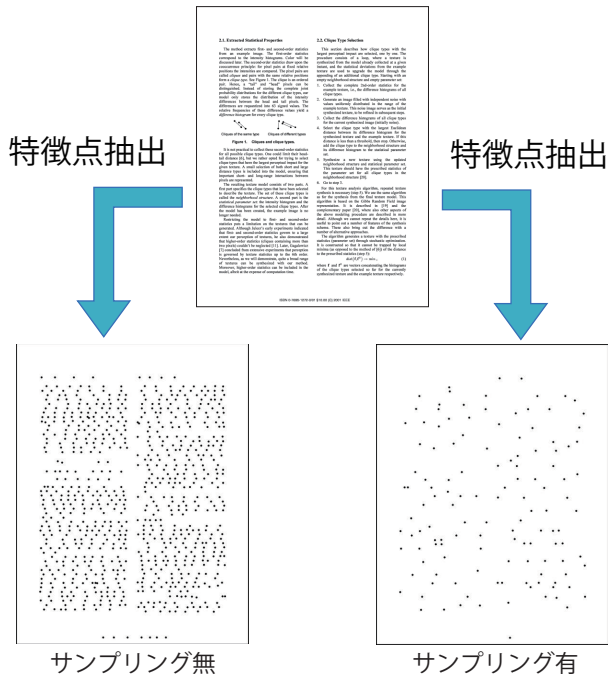


図 5 特徴点のサンプリング

サンプリングを行った結果を図 5 に示す。左下にサンプリングを行わない場合、右下にサンプリングを行った場合の特徴点抽出結果を示す。この例で示すように、サンプリングされた特徴点は文書全体に分布しているの、上記の要求を満たすことができる。特徴量はサンプリングされた特徴点の組合せを計算するのではなく、従来手法と同様に近傍点を用いて計算される。ここで、サンプリングされた特徴点の近傍点との距離について考える。面積の小さい単語領域から抽出された特徴点は、近傍点との距離が小さくなる。近傍点との距離が小さいほど射影歪みの影響を受けにくいため、より安定した特徴量計算が行える。

ただし、上記の手法でサンプリングされた特徴点は数が少ないことが予備実験からわかっている。そこで、サンプリング特徴点の近傍 k 点もサンプリングする。 k の値は、1 文書あたり 200 点がハッシュに保存されるよう、文書ごとに設定する。特徴点数が 200 点に満たない文書に対してはサンプリングしない。このように、サンプリングする文書としない文書に分かれる。しかし、検索時にサンプリングの有無を判別するのは困難である。そこで、検索では得られたすべての特徴点についてハッシュを参照する。

3.2 特徴量の識別性・安定性向上

一般的に、データベースの大規模化に伴い、検索精度が低下する。それは LLAH においても同様で、登録ページ数を増加させると検索精度が低下することがわかっている [10]。これは、類似した特徴点の配置を持つ文書が登録される可能性が高くなるためである。これにより、誤投票が増加するため、検索精度が低下する。また、ハッ

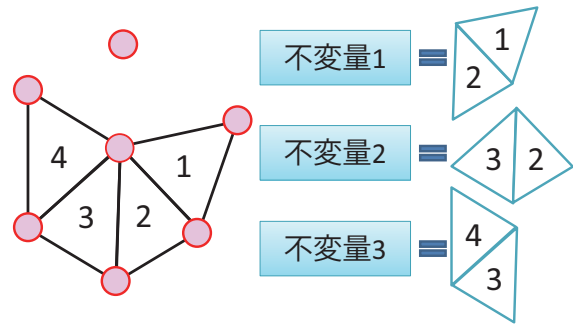


図 6 不変量の相関関係

シュに保存される特徴量が膨大になるため、衝突回数が増加する [10]。その結果、リストを参照する時間が増え、処理時間が増大すると考えられる。以上の問題を解決するため、特徴量の識別性を高める必要がある。

識別性を向上させるため、特徴量の次元数を増加させる。次元数を増加させる方法として、別の新たな特徴量を追加するというのが考えられる。しかし、この方法では新たな特徴量の計算により処理時間が長くなる。そこで、アフィン不変量の数を増やすことで次元数を増加させる。具体的には、パラメータ n, m の値を大きくする。オリジナルの LLAH では、 $n = 7, m = 6$ と設定しているため、21 次元の特徴量を得ている。従来ではデータベースの大規模化を想定していないため、これで十分な識別性を持つ。提案手法では、 $n = 8, m = 7$ とそれらの値を大きく設定する。これにより、アフィン不変量ベクトルが ${}_{7}C_{4} = 35$ 次元、面積比特徴量が 7 次元となり、合計 42 次元の特徴量を得ることができる。

しかし、次元数の増加によって識別性が向上するのに対して、特徴量の安定性は低下する。検索では、特徴量のアフィン不変量がすべて一致する必要がある。単語領域の重心は比較的安定して抽出できるが、少なからず位置の変動が起こる。これによりアフィン不変量には誤差が生じる。次元数が増加すると、一致しない値が得られる可能性が高くなる。これが安定性低下の理由である。

この問題を解決するため、冗長な次元を削除することを考える。従来手法では、特徴量計算において同一の三角形を重複して用いている。例えば図 6 に示すように、不変量 2 は不変量 1 と 2 番の三角形を、不変量 3 とは 3 番の三角形を重複して使用する。このため、不変量 2 は不変量 1, 3 と相関が高く、冗長であると考えられる。そこで、不変量が同一の三角形を共有しないように不変量を計算することで、特徴量の冗長性を解消する。これにより、特徴量はアフィン不変量ベクトル 17 次元、面積比特徴量 7 次元の計 24 次元となる。

4. 実験

4.1 実験 1：提案手法の効果

本稿で述べた LLAH の改善法の効果を検証するため

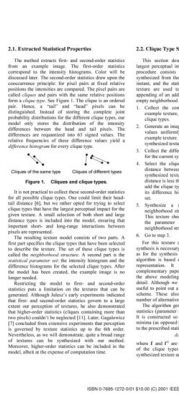


図 7 登録文書画像

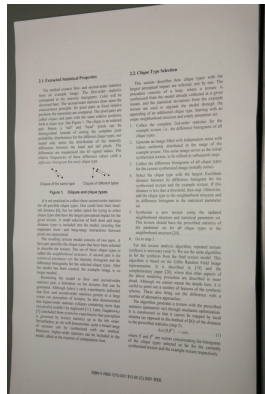


図 8 検索質問画像

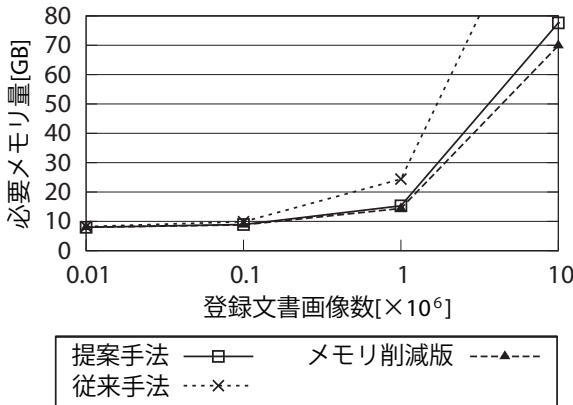


図 9 登録ページ数と必要メモリ量の関係

に、従来手法と提案手法を用いて実験を行った。具体的には、以下の3つのバージョンのLLAHでの必要メモリ量、検索精度、処理時間を調べることにより、改善効果を調べた。

- (1) 従来手法
- (2) メモリ削減版(次元数増加なし)
- (3) 提案手法

実験のために、登録文書画像数の異なる4つの文書画像データベースを作成した。登録文書画像数はそれぞれ、1万、10万、100万、1000万ページである。登録文書画像数の異なるデータベースを用いて実験を行うことで、それぞれの手法のスケラビリティを検証した。登録文書画像は、インターネットから収集したPDFファイルを200dpiで画像に変換したものである。登録文書画像の例を図7に示す。検索質問は、印刷文書全体が写るように紙面に対して斜め方向(正面を90°としたときに約60°)から撮影した画像を用いた。画像枚数は1003枚である。撮影に用いたデジタルカメラは、1200万画素のものである。検索質問画像の例を図8に示す。登録画像(90°)と検索質問(60°)の撮影角度が異なる理由は、射影歪みに対するロバスト性を確認するためである。必要メモリ量は、検索に必要なすべてのメモリ使用量を表す。また、処理時間は検索質問画像1枚当たりの平均であり、特徴点抽出にかかる時間は含まれない。ハッシュ表のサ

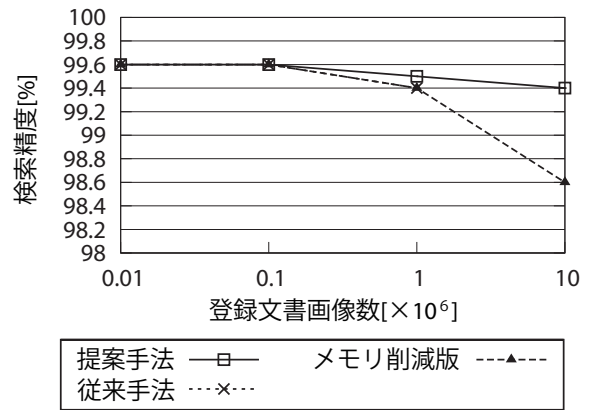


図 10 登録ページ数と検索精度の関係

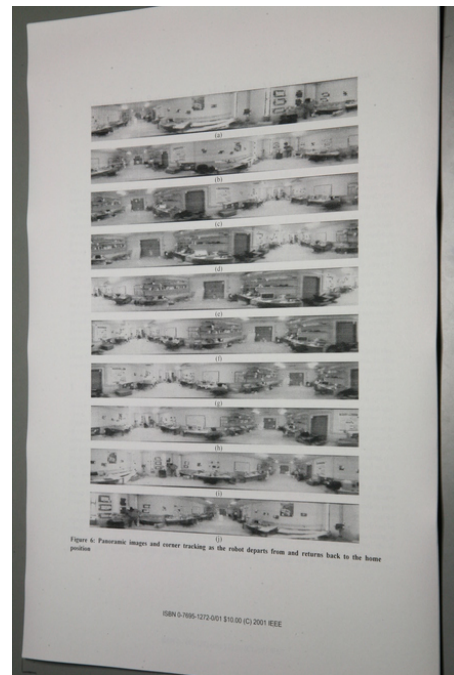


図 11 検索失敗例

イズは $2^{30} - 1$ である。実験に用いた計算機は、CPUがAMD Opteron 2.8GHz、メモリが128GBのものである。

4.1.1 必要メモリ量

登録ページ数と必要メモリ量の関係を図9に示す。ここで、ハッシュ表を確保するためにメモリを8GB使用している。本実験で用いた計算機では、従来手法における1,000万ページのデータベースは作成不可能である。従って、従来手法での1,000万の結果は推定値である。

100万ページでの結果を見ると、メモリ削減版のLLAHは必要メモリ量を約50%削減したことがわかる。一方で、提案手法の必要メモリ量はメモリ削減版と比較して大きくなっている。これは、1つの特徴点から計算される特徴量の数の違いによるものである。具体的には、メモリ削減版が ${}^7C_6 = 7$ 個の特徴量を計算するのに対し、提案手法では ${}^8C_7 = 8$ 個の特徴量を計算する。このように、提案手法ではデータベースに登録する特徴量数が多いため、メモリ使用量の増加につながったと考えられる。

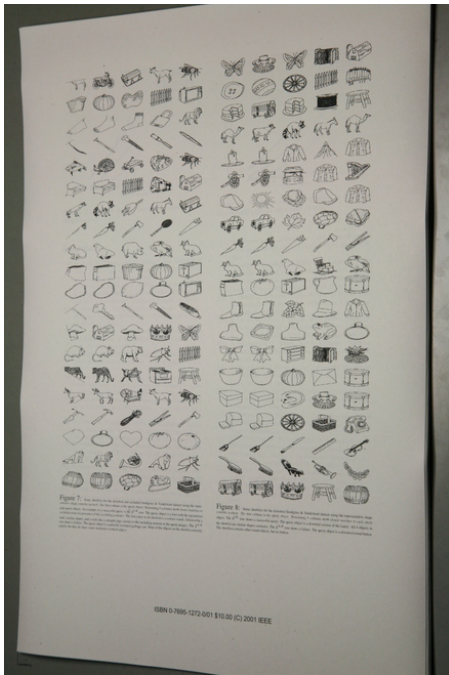


図 12 大規模化による検索失敗

ただし、この増分を考慮しても、提案手法において大幅なメモリ削減が実現できたといえる。提案手法での1000万ページにおける必要メモリ量は約77GBであった。

4.1.2 検索精度

登録ページ数と検索精度の関係を図10に示す。1000万ページのデータベースにおいて、提案手法の検索精度は99.4%と他の2手法と比較して高い性能を示した。提案手法では、検索質問画像1枚当たりの誤投票数が144票あった。一方、メモリ削減版での誤投票数は443票と、提案手法の3倍以上となった。このように、特徴量の識別性を向上させたことによって誤投票の数を削減することができ、その結果高い検索精度を実現することができたと考えられる。

1万ページにおいて検索に失敗した例を図11に示す。この例を見ると、文書の大半が図表であり、テキストの量が少ないことがわかる。現在の手法では単語領域の重心を特徴点とするため、この画像から得られる特徴点数は非常に少ない。そのため、正解の文書が十分な得票数を得られず、検索に失敗したと考えられる。

大規模化に伴い検索に失敗した例を図12に示す。この例においても大半が図であり、安定した特徴点が抽出できない。また、図が大量にあり、特徴点が多く得られることから、大規模化に伴い誤投票が増加する。その結果、検索できなくなったと考えられる。

以上で示したように、文書の多くを画像が占めるような場合には、高精度な検索が行えない。この問題を解決する方法として、文献[2]で提案されているような高速な画像認識の手法を取り入れることが考えられる。

4.1.3 処理時間

登録ページ数と処理時間の関係を図13に示す。提案

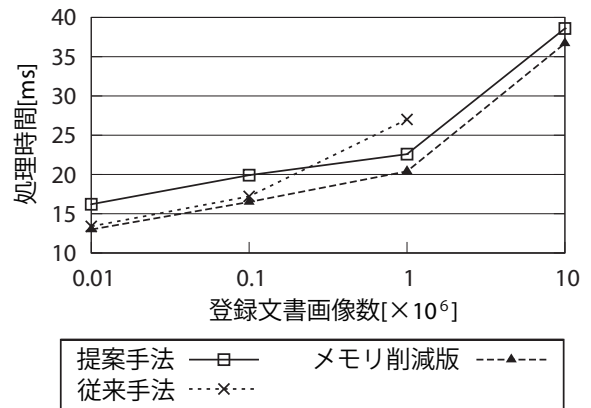


図 13 登録ページ数と処理時間の関係

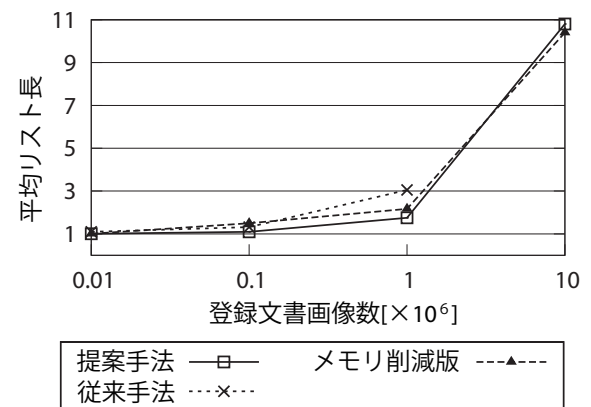


図 14 登録ページ数と平均リスト長の関係

手法の処理時間は、1000万ページデータベースにおいて38msと、実時間検索可能な値となった。また、登録ページ数の桁上がりと比較して、処理時間の増加は抑えられている。これは、ハッシュによる効果であると考えられる。

登録文書画像数が少ない場合、提案手法の方が従来手法よりも処理時間が長い。これは、各手法における計算量が異なるためである。従来手法では、特徴点あたり ${}_{7}C_{6} = 7$ 個の特徴量と、特徴量あたり ${}_{6}C_{4} = 15$ 個の不変量を計算する。一方、提案手法では、 ${}_{8}C_{7} = 8$ 個の特徴量と、24個の不変量を計算する。このように、提案手法の方が従来手法よりも計算量が多いため、処理時間が長くなったと考えられる。

一方、登録文書画像数が多い場合、提案手法の方が高い性能を示した。これは、ハッシュ表におけるリスト長の違いが原因であると考えられる。ここで、平均リスト長を図14に示す。平均リスト長とは、ハッシュ表においてリスト長が0でないものの平均を表す。データベースが大規模になるにつれ、平均リスト長が増加しているが、提案手法のリスト長の増加は抑えられている。100万ページにおいて提案手法のリスト長は、従来手法の約1/2となっている。これは、特徴量の識別性を向上させたことによる効果であると考えられる。従来手法のようにリスト長が大きくなると、リストをたどる処理時間が

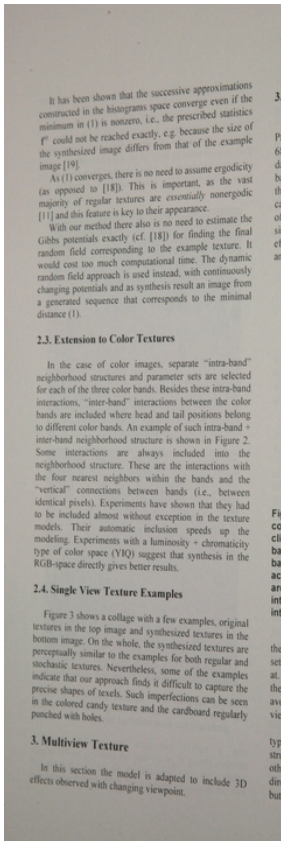


図 15 撮影範囲：1/2

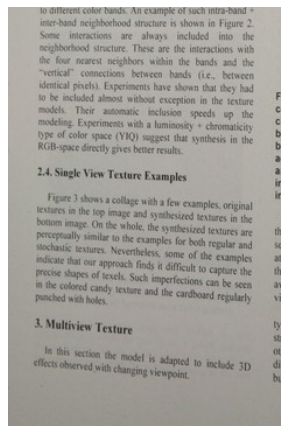


図 16 撮影範囲：1/4

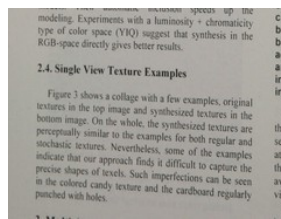


図 17 撮影範囲：1/8

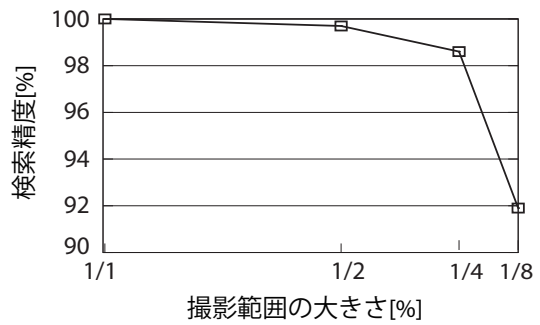


図 18 撮影範囲による検索精度の違い

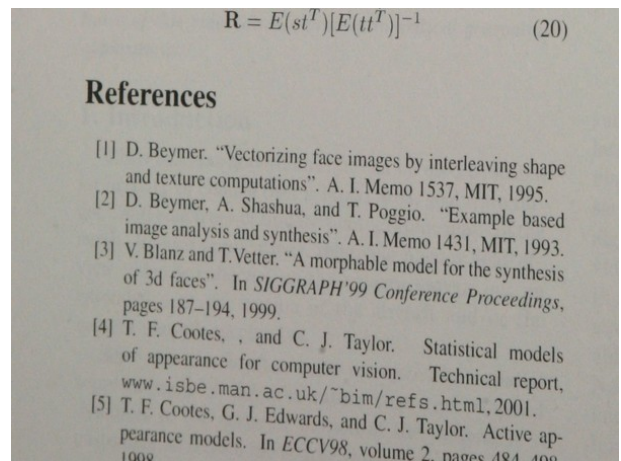


図 19 部分撮影での検索失敗例

増加する．そのため，提案手法の方が高い性能を示したと考えられる．ここで，1000 万ページにおいて平均リスト長が急激に増加しているが，これはハッシュテーブルが飽和状態になったためである．提案手法では，すべてのハッシュのピンのうち 99.6% が使用されている状態であった．このことから，さらなる大規模化を行うにはより大きなハッシュテーブルを用意する必要がある．

4.2 実験 2：部分撮影への耐性

オリジナルの LLAH ではすべての特徴点を用いてインデキシングしていたのに対し，提案手法ではサンプリングされた特徴点しか保持していない．従って，検索質問画像から得られた特徴点の数が十分でない場合，提案手法はオリジナルの LLAH よりも低い性能を示す可能性がある．特徴点数が十分でないような状態は，文書中の狭い領域を撮影したときに生じると考えられる．そこで本実験では，提案手法における撮影範囲の大きさと精度の関係を調べる．

実験で用いるデータベースには，1,000 万ページの文書画像が登録されている．本実験では，実験 1 で用いた検索質問画像から様々な大きさと切り抜くことによって，撮影範囲の大きさが異なる検索質問を作成した．まず，実験 1 で用いた 1003 枚の検索質問から，大部分が図表であるものを除いた 989 枚を選択した．それらから文章が写っている部分を切り抜いたものを本実験での検索質問とした．切り抜く大きさは，全体の 1/2, 1/4, 1/8 で

ある．検索質問の例を，それぞれ図 15, 16, 17 に示す．結果を表 18 に示す．撮影範囲が狭くなるに伴い，検索精度が低下することがわかる．撮影範囲の小さな検索質問からは得られる特徴点数が少ない．従って，正解の文書が十分な得票数を得られず，検索に失敗したと考えられる．

検索失敗した例を図 19 に示す．このように，文書中に点(「.」)やクォーテーションマーク(「”」)が多く分布していると，検索に失敗する傾向があった．これは，それらがノイズとなって，安定した特徴点を抽出することが困難であるためである．図 20, 21 にその例を示す．図 20 では，検索質問画像において「”」が単語から切り離されて，登録文書画像にはない特徴点が抽出されている．また図 21 では，登録文書画像では抽出されていた「.」からの特徴点が，検索質問画像では抽出されなかった．このような特徴点の出現や消失といった現象が起こりやすいため，安定した特徴点抽出が行えず検索精度が低下したと考えられる．これは，特徴点抽出処理における適応二値化によって引き起こされる．これについては，特徴点抽出に新たな処理を追加する等，改良が必要である．

しかし，検索精度についてみれば，1/8 の大きさであっても 92% ，1/4 の大きさであれば 98% 以上と非常に高い精度を保っている．従って，提案手法は撮影範囲の大きさに対してロバストであると考えられる．

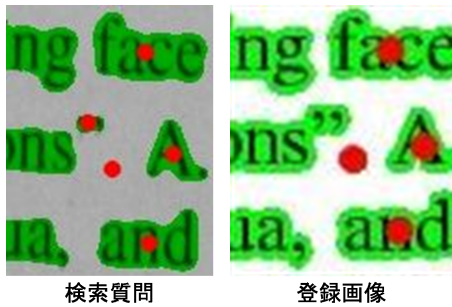


図 20 特徴点の出現

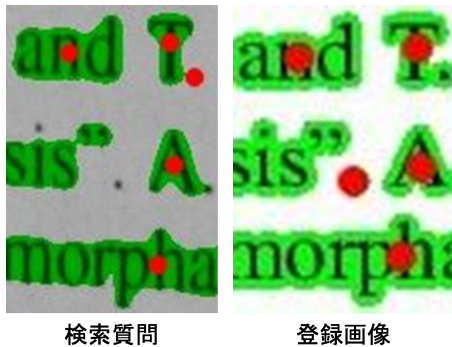


図 21 特徴点の消失

表 1 2000 万ページデータベース

必要メモリ量 [GB]	検索精度 [%]	処理時間 [ms]
150	99.3	54.5

4.3 2000 万ページデータベース

最後に、さらなる大規模化の足掛かりとして、提案手法を用いた 2000 万ページのデータベースでの検索を行った。検索質問としては実験 1 と同じものを用いた。実験 1 の結果から、1000 万ページでハッシュ表が飽和状態であったため、ハッシュ表のサイズを 2 倍、即ち $2^{31} - 1$ (16GB) とした。実験に用いた計算機は、CPU が AMD Opteron 2.2GHz、メモリが 256GB のものである。この計算機は、前の実験に用いたものよりメモリは大きい、速度は 85% 程度である。

結果を表 1 に示す。必要メモリ量は 150GB と膨大になったが、99.3% という高い検索精度を実現した。また、処理時間に関しても 54.5ms と良好な結果を得た。この結果は、2000 万ページという大規模なデータベースでの実時間検索が可能であることを示している。実験 1 の 1000 万ページデータベースでの処理時間と比較すると大きく増加しているが、これは本実験で用いた計算機の CPU が実験 1 で用いたものよりも低速であるためである。以上の結果から、提案手法は 2000 万ページデータベースでの高精度な実時間検索を実現可能であることが確かめられた。

5. まとめ

本稿では、LLAH の改良するための手法を 3 つ提案し

た。特徴点のサンプリングによる必要メモリ量の削減を実現した。また、特徴量の次元数増加と冗長性の除去により、識別性・安定性を向上させることができた。実験により、必要メモリ量を 50% 削減し、2,000 万ページデータベースにおいて精度 99.3%、処理時間 54.5ms で検索可能であることが確かめられた。また、撮影範囲の大きさへの耐性を検証した結果、文書画像の 1/4 の大きさあっても 98% 以上の検索精度となることが確認された。

今後の課題としては、さらに必要メモリ量を削減することが挙げられる。また、より省メモリな特徴量を考案することも必要である。さらに、より安定な特徴点抽出法を考案する必要もある。

謝 辞

本研究を遂行する上で、東北大学大町真一郎教授、九州大学内田誠一教授から有益なコメントを頂いた。本研究の一部は、科学技術振興機構 CREST、ならびに日本学術振興会科学研究費補助金基盤研究 (B) (22300062) の補助による。

文 献

- [1] “<http://www.google.com/mobile/goggles/>”
- [2] 野口和人, 黄瀬浩一, 岩村雅一, “近似最近傍探索の多段階化による高速特定物認識,” 信学論 (D), vol. J92-D, no. 12, pp. 2238–2248, Dec. 2009.
- [3] D. Doermann, “The indexing and retrieval of document images: A survey,” Computer Vision and Image Understanding: CVIU, vol. 70, no. 3, pp. 287–298, June 1998.
- [4] D. Doermann, H. Li, O. Kia, “The Detection of Duplicates in Document Image Databases,” Proc. 4th International Conference on Document Analysis and Recognition (ICDAR1997), pp. 314–318, 1997.
- [5] P. Clark, M. Mirmehdi, “Recognizing text in real scenes,” International Journal on Document Analysis and Recognition, vol. 4, no. 4, pp. 243–257, Aug. 2002.
- [6] 中居友弘, 黄瀬浩一, 岩村雅一, “Web カメラを用いたリアルタイム文書画像検索,” 信学論 (D), vol. J90-D, no. 8, pp. 2262–2265, Jan. 2007.
- [7] T. Nakai, K. Kise, M. Iwamura, “Real-Time Retrieval for Images of Documents in Various Languages using a Web Camera,” Proc. 10th International Conference on Document Analysis and Recognition (ICDAR2009), pp. 146–150, Beijing, China, Jul. 2009.
- [8] K. Kise, M. Chikano, K. Iwata, M. Iwamura, S. Uchida, S. Omachi, “Expansion of Queries and Databases for Improving the Retrieval Accuracy of Document Portions — An Application to a Camera-Pen System,” Proc. 9th IAPR International Workshop on Document Analysis Systems (DAS2010), pp. 309–316, Boston, MA, USA, Jun. 2010.
- [9] 中居友弘, 黄瀬浩一, 岩村雅一, “特徴点の局所的配置に基づくリアルタイム文書画像検索とその拡張現実への応用,” 信学技報, vol. 106, no. PRMU-229, pp. 41–48, Sept. 2006.
- [10] 中居友弘, 黄瀬浩一, 岩村雅一, “デジタルカメラを用いた高速文書画像検索におけるアフィン不変量および相似不変量の利用,” 信学技報, vol. 105, no. PRMU-614, pp. 25–30, Feb. 2006.