

マルチメディアデータ検索における情報検索エージェント (セマンティックギャップを克服するシステムモデル)

世良田 学

Abstract

A new query method, called query by visual keys, is proposed to aim easy operation and efficient region-based image retrieval (RBIR). Visual keys are constructed from representative regions/subimages in a given image database, and the database is indexed with visual keys. A system on PC is presented, where text retrieval techniques are applied to the image retrieval with visual keys. Experimental results show that one retrieval is done within 4ms and that the proposed system achieves the comparable retrieval precision (with user-friendly operation and low computational cost) to conventional region based image retrieval systems.

1 はじめに

現在、膨大なマルチメディアデータを有し、さらに年々コンテンツ数が増加の一途を辿る WWW(the World Wide Web) を一つの大きな知識情報源として有効利用することに対する需要は、WWW に対する情報収集・検索エージェントの研究が盛んに行われていることから明らかである。21 世紀 COE プログラム「エージェントベース社会システム科学の創出」におけるプロジェクト「意思決定の数理モデル」の研究の一環として、「マルチメディアデータ検索における情報検索エージェント」に取り組む。ただし、WWW 上のデータについて、現在、大半がテキストデータと画像データであると考え、まずは今後需要が増していくであろう画像データ検索に焦点をあて、マルチメディア検索の一部として研究を行う。

画像検索におけるクエリ方式は大きく 2 つ(キーワード、画像) に分類される。キーワードによる検索はユーザに使いやすいクエリ作成インタフェースと効率的な検索によるレスポンスの良さを提供する反面、画像データ自体の考慮が容易ではない。また、画像による検索はクエリ作成の負担は小さくないものの、画像データ自体を考慮した検索を行う。ただし、どちらにおいてもユーザモデルとシステムモデルとの間にセマンティックギャップというものが生じる。本研究では、そのセマンティックギャップを縮め、また、ユーザにとって使い勝手の良い検索システムを提供するために、ビジュアルキーという概念を提案し、実験によって性能を検証する。

2 ビジュアルキーを用いた画像検索システム

2.1 画像検索における 2 つのクエリ方式

既存の画像検索システムは、クエリ方式によって大きく 2 つ(キーワード、画像) に分類される。

google のイメージサーチが最も知られているキーワードをクエリに用いる手法は、ユーザに使いやすいクエリ作成インタフェースを提供し、また、システムは画像をキーワードで索引付けることでテキスト検索のように高速で効率の良い検索を可能にしている反面、画像データ自体の考慮が容易ではない。一方、画像(特徴量) をクエリに用いる手法は、多くのユーザにとってクエリ作成の負担は小さくなく、検索効率も高くはないが、画像データに基づいて検索を行うことから、CBIR(content-based image retrieval) と呼ばれ、IBM の QBIC[5] や MIT の Photobook [11][12][13] などが有名なものとしてある。

2.2 セマンティックギャップ

前節で述べた 2 つのクエリ手法それぞれに関わらず、ユーザの検索質問に対するシステムの出力結果はユーザの意図したものと異なることがある。それは、ユーザが検索要求をクエリとして十分に記述できないことや、ユーザをモデル化しシステムに組み込む際に生じるユーザモデルとシステムモデルとのギャップによるものである。後者のギャップをセマンティックギャップといい、このギャップを縮めるための研究が盛んに行われている。

2.3 ビジュアルキーシステムの提案

2.1 節で述べた 2 つのクエリ方式には得手不得手がある。そこで、画像・テキストデータ両方を有する WWW 上から、キーワードを関連付けた代表的部分画像を生成することで、ユーザにキーワードを主体とし、必要に応じて画像データを指定する使い勝手の良いクエリ方式を提供し、システムは画像データを考慮した検索を行うことを目指す。

また、画像理解の研究が発展途上にあることから、システムモデルが画像データのみから”類似している”と判断するものは、意味的なものを考慮するユーザのモデルの判断とはギャップが生じることもある。そこで、テキストデータによってシステムの画像内容理解を補助することで、よりユーザモデルに近いシステムモデルを構築することを目標とする。

3 ビジュアルキー

3.1 ビジュアルキーの概念

キーワードを関連付けられた代表的部分画像をビジュアルキーとして定義するが、本研究では、まず、テキストデータを除いた代表的部分画像をビジュアルキーとして取り扱う。

提案システムの概要を図 1 に示す。

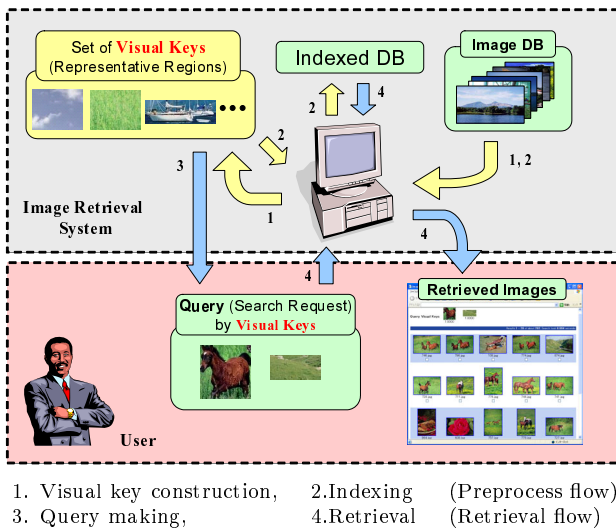


Figure 1: 提案システムの概要

3.2 ビジュアルキー生成

ビジュアルキーの生成は、図 2 で示される 3 つのパートからなる。

1. 画像分割
2. 特徴量抽出
3. 部分画像分類とビジュアルキー選定

3.2.1 画像分割

ターゲットとなるデータベース DB から全部分画像集合 $S(= \{s_{ij} | s_{ij} \in S(I_i), \forall I_i \in DB\})$ を得るために、データベース DB 中の全ての画像 I_i を部分画像集合 $S(I_i)(= \{s_{(i,j)}\})$ に分割する。ここで $s_{(i,j)}$ とは、画像 I_i からセグメンテーションによって得られる j 番目の部分画像 $s'_{(i,j)}$ を内接する矩形の画像である。セグメンテーションは目的に応じて様々な手法があるが、同じ画像検索を目的にした手法であり、かつ、実際の分割結果も適切なものが得られていると感じた Blobworld system [6] の手法を提案システムに適用した。

3.2.2 特徴量抽出

得られた全部分画像集合 S の各部分画像 $s_{(i,j)}$ に対し、内包する部分画像 $s'_{(i,j)}$ から以下で説明する color moments (9 次元), texture neighborhood (8 次元), そして Sobel edge detector (8 次元) の 25 次元の特徴ベクトルを抽出する。

Color Moments: 部分画像から抽出する色特徴として color moments[2] を用いる。この特徴はヒストグラムに基づく手法によるものに比べ、検索速度は速く、結果も頑強なものである。この color moment は対象部分画像内における各色チャンネルの値の平均、標準偏差、歪度を算出します。部分画像 s'_k ($k = (i, j), s'_k \subset s_k$) 内の m チャンネルにおける n 番目の画素値を p_k^{mn} 、そしてその部分画像 s'_k の画素数を N_k とした時、 m チャンネルにおける color moments (平均, 標準偏差, 歪度) は以下のように定義される。

$$\mu_k^m = \frac{1}{N_k} \sum_{n=1}^{N_k} p_k^{mn}, \quad (1)$$

$$\sigma_k^m = \left(\frac{1}{N_k} \sum_{n=1}^{N_k} (p_k^{mn} - \mu_k^m)^2 \right)^{\frac{1}{2}}, \quad (2)$$

$$\gamma_k^m = \left(\frac{1}{N_k} \sum_{n=1}^{N_k} (p_k^{mn} - \mu_k^m)^3 \right)^{\frac{1}{3}}, \quad (3)$$

提案手法では、L*a*b*表色系における color moments を用いた。L*a*b*表色系は人間の心理 4 原色 (緑、赤、青、黄) に基づいた表色系であり、その表色系空間内の複数の座標点の距離感覚を比較すると人間の感覚に近いことが特徴である。

この色特徴は $3 \times 3 = 9$ 次元ベクトル $\vec{c}_k (= [\mu_k^L, \mu_k^a, \mu_k^b, \sigma_k^L, \sigma_k^a, \sigma_k^b, \gamma_k^L, \gamma_k^a, \gamma_k^b])$ になる。

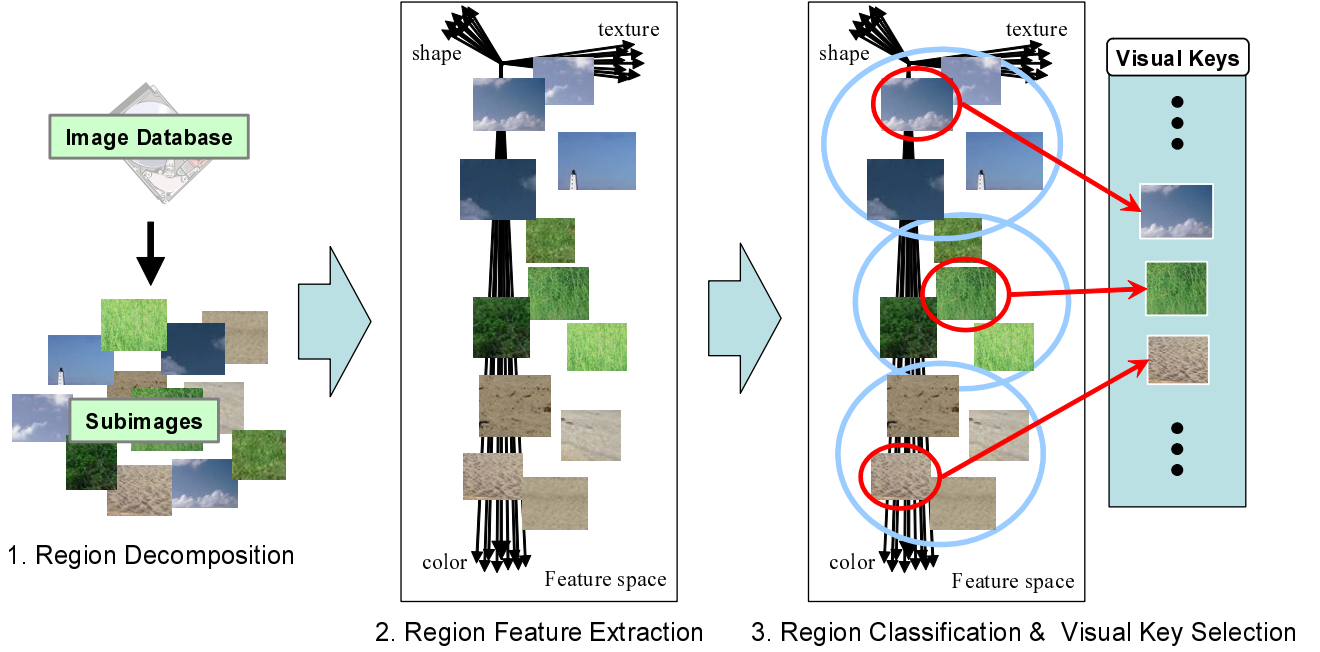


Figure 2: ビジュアルキー生成の流れ

Texture Neighborhood: texture neighborhood[3] を部分画像のテクスチャ特徴として用いる。部分画像 s'_k における L^* (luminance) チャンネルにおいて、全画素と8近傍画素を比較し、ある画素に対しそれぞれの近傍の画素値がその画素の値より大きくなる確率を求める。

$$t_k^d = \frac{1}{N_k} \sum_{n=1}^{N_k} TH(L_k^{nd}, L_k^n + \delta_L), \quad (4)$$

$$TH(a, b) = \begin{cases} 1 & \text{if } a > b, \\ 0 & \text{otherwise,} \end{cases}$$

ここで、 $L_k^{n(d)} (\in [0, \dots, 1])$ は、部分画像 s'_k の n 番目の画素における (方向 d の画素の) 輝度値で、 $d = 1, \dots, 8$ 、そして $\delta_L (= 0.015)$ を閾値とする。そこから得られる8次元ベクトル $\vec{t}_k (= [t_k^1, \dots, t_k^8], t_k^d \in [0, 1])$ を部分画像 s_k のテクスチャ特徴として用いる。

Sobel Edge Detector: 形状特徴として Sobel edge detector [4] を以下のように計算する。まず、画像を HSI 空間に変換し、色相 (H:hue) チャンネルを除き他の2チャンネルに対して8つの Sobel 演算を行う (図 3)。

Sobel 演算を適用した画像を各チャンネルで閾値 e_{km}^{dn} を用いて2値画像に変換する。部分画像 s'_k の n 番目の画素の m 番目の色チャンネルにおける方向 $d (\in \{1, \dots, 8\})$ の Sobel 演算値を g_{km}^{dn} とおき、2値化された値は以下のように得られる。

$$e_{km}^{dn} = TH(g_{km}^{dn}, r_m \max_{d,n} g_{km}^{dn}), \quad (5)$$

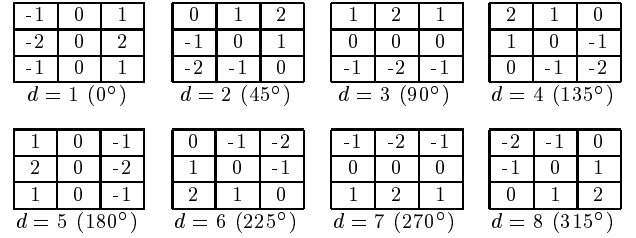


Figure 3: Sobel 演算と対応する傾き方向 d .

ここで、閾値 r_I と r_S はそれぞれ 0.15 と 0.35 となる。その色の強さ (I:intensity) と彩度 (S:saturation) の2値化画像を論理和によって統合する。そして、以下のようにそれぞれの Sobel 演算の向きにおいて、検出されたエッジ画素をカウントすることで、8次元のエッジヒストグラム $\vec{s}_k (= [s_k^1, \dots, s_k^8], s_k^d \in [0, 1])$ を得る。

$$s_k^d = \frac{1}{N_k} \sum_{n=1}^{N_k} e_k^{dn}, \quad (6)$$

この時、 e_k^{dn} は e_{kI}^{dn} と e_{kS}^{dn} の論理和を取った結果である。

部分画像 s_k の特徴ベクトルは上で述べた各特長量からなる 25 次元の特徴ベクトル $\vec{f}_k = [\vec{c}_k, \vec{t}_k, \vec{s}_k]$ となる。得られた 25 次元の特徴空間において、2つの特徴ベクトル \vec{f}_k と \vec{f}_l 間の距離 $d(\vec{f}_k, \vec{f}_l)$ を以下のようにとる。

$$d(\vec{f}_k, \vec{f}_l) = d_c(\vec{c}_k, \vec{c}_l) + \|\vec{t}_k - \vec{t}_l\| + \|\vec{s}_k - \vec{s}_l\|, \quad (7)$$

$$d_c(\vec{c}_k, \vec{c}_l) = \sum_{m \in \{H, S, V\}} w_{m1} |\mu_k^m - \mu_l^m| + w_{m2} |\sigma_k^m - \sigma_l^m| + w_{m3} |\gamma_k^m - \gamma_l^m|, \quad (8)$$

ここで、 $w_{L1}, w_{L2}, w_{L3} = 0.5$, $w_{a1}, w_{a2}, w_{a3} = 0.5$, and $w_{b1}, w_{b2}, w_{b3} = 0.25$ を経験的に選択した。

3.2.3 部分画像分類とビジュアルキー選定

得られた特徴ベクトル群 $\{\vec{f}_i : 1 \leq i \leq |S|\}$ を階層的クラスタリングアルゴリズムによって c_{vk} 個のクラスタに分類する。

クラスタ初期化として、式 (1)-(6) で得られた特徴ベクトルに対応付けられた $|S|$ 個のクラスタを用意する。そして、クラスタを統合するかどうかの尺度として、クラスタ C_k と C_l 間で以下の計算を行い、最も値が小さかったクラスタ同士を統合していく。

$$D(C_k, C_l) = \frac{|C_k||C_l|}{|C_k| + |C_l|} d(\vec{f}_{\hat{C}_k}, \vec{f}_{\hat{C}_l}), \quad (9)$$

ここで、 $|C_k|$ は C_k 内の要素 (特徴ベクトル) 数で、 $\vec{f}_{\hat{C}_k}$ は C_k の重心ベクトルである。

それぞれのクラスタにおいて、重心ベクトルに最も近い特徴ベクトルを持つ部分画像をビジュアルキーとして選定する。得られた全ビジュアルキー集合を $\mathbf{VK} = \{vk_u \in S : 1 \leq u \leq c_{vk}\}$ と書く。また、最適な分類クラスタ数 (ビジュアルキーの数) は目標データベースに応じて変化させる。ここでは c_{vk} を検索性能の最適処理を通して得た。(5.2 節参照)

4 ビジュアルキーを用いた画像検索システム

4.1 索引付け

ビジュアルキーに基づいた検索方式は、キーワードを用いたテキスト検索のそれに似ていることから、テキスト検索分野の技術を容易に適用可能である。提案システムでは TFIDF (term frequency - inverse document frequency) [22] と転置インデックス法 [24] を用いる。TFIDF は各ビジュアルキーに重要度を設けることを可能とし、転置インデックス法は検索速度の向上を可能とする。

対象データベース中の画像のインデックスは以下のように構成する。

1. 部分画像抽出

対象となる画像データベース DB の各画像 I_i から部分画像集合 $S(I_i)$ を 3.2.1 節で用いたセグメンテーションを用いて同様に抽出する。

2. ビジュアルキーによる表現

画像 I_i から得られた部分画像集合 $S(I_i) = \{s_{ij}\}$ の各部分画像 s_{ij} を以下のように特徴空間において最近傍にあるビジュアルキーに対応付ける。

$$\mathbf{VK}(I_i) = \left\{ vk_v \left| vk_v = \operatorname{argmin}_{vk_u \in \mathbf{VK}} d(\vec{f}_{s_{ij}}, \vec{f}_{vk_u}), s_{ij} \in S(I_i) \right. \right\}, \quad (10)$$

ここで、 $d(\vec{f}_{s_{ij}}, \vec{f}_{vk_u})$ は式 (7) で説明したものである。

3. インデックスの構築

それぞれのビジュアルキー vk_u に対し、以下のようにインデックスを構築する。

$$id(vk_u) = \{ (I_i, d(\vec{f}_{s_{ij}}, \vec{f}_{vk_u})) | I_i \in \mathbf{DB}, vk_u \in \mathbf{VK}(I_i), s_{ij}: \text{corresponding region of } vk_u \text{ (in } I_i) \}. \quad (11)$$

画像データベースにおける各ビジュアルキーの出現頻度には差があることから、各ビジュアルキーの重要度を出現頻度で特徴付けるために、テキスト検索関連技術の TFIDF [22] を用いる。TF (Term Frequency) と IDF (Inverse Document Frequency) は各ビジュアルキーの画像における局所的な重みと画像データベースにおける大局的な重みとして以下のように算出する。

$$TF_{ui} = \begin{cases} 0.5 + 0.5 \frac{f_{ui}}{\max_v f_{vi}} & (f_{ui} > 0) \\ 0 & (f_{ui} = 0) \end{cases}, \quad (12)$$

$$IDF_u = \frac{1}{n_u}, \quad (13)$$

ここで、 f_{ui} は画像 I_1 におけるビジュアルキー vk_u に対応する部分画像の百分率であり、 $n_u (> 0)$ はデータベース中のビジュアルキー vk_u に対応する部分画像を含んでいる画像の数である。

4.2 画像検索

4.2.1 ビジュアルキーを用いたクエリ表現

提案画像検索システムにおいて、ユーザはクエリ $q = \{vk\}$ としていくつかのビジュアルキーをシステムから与えられたビジュアルキー集合 \mathbf{VK} の中から指示する。ここで、それぞれの vk はユーザが望んだ画像をシステムが探すためのヒントとして扱う。

4.2.2 クエリマッチング

与えられたクエリ $q (= \{vk\})$ に対し、提案システムは以下のように適合する画像を選択する。

$$\mathbf{M}(q) = \{I_i | (I_i, d) \in id(vk), vk \in q\}. \quad (14)$$

$M(q)$ 内の各画像は以下のようにして得られる値に応じて順位付けを行う。

$$Q(I_i) = \sum_{(I_i, d) \in \text{Uid}(vk_u)} w_u TF_{ui} IDF_u (1 - d),$$

for $\forall I_i \in M(q)$, (15)

$$w_u = \begin{cases} 1 & (vk_u \in q) \\ 0 & (\text{otherwise}) \end{cases}$$

ここで、 w_u は 0 から 1 の値をとり、今後、クエリとして与えられたキーワードに応じて重み付けられる際や適合フィードバック??を適用する際に用いる。1 に近いほど重要で、0 に近いほど重要ではなくなるとみなし、マッチした画像 $M(q)$ 全てを上式の値 $Q(I_i)$ で順序付ける。

5 COREL を用いた検索実験

提案システムを Java 言語で Pentium 4 (1.5GHz) PC 上に実装し、COREL データベースから選定した画像セットを利用して実験を行った。それぞれの画像において、3.2.1 で述べた特徴量などの情報は事前に算出・記録しておく。

5.1 実験条件

システムを COREL データベース中の 1000 枚の画像において評価する。その画像は 10 の画像カテゴリ (Africa, Beach, Buildings, Buses, Dinosaurs, Elephants, Flowers, Horses, Mountains, Food) に分類されており、それぞれのカテゴリにおいて 100 枚の画像がある。また、その 1000 枚の画像からは、3.2.1 節で述べたセグメンテーションによって 8,962 の部分画像が得られる。検索性能実験は、定性的な評価実験ではなく、定量的な評価実験として、1000 枚の画像をそれぞれユーザの検索要求としてシステムにクエリとして与え、クエリ画像と検索結果の画像が同じ COREL 画像カテゴリであれば適合しているとみなし以下で説明する値を評価に用いる。

$ID(i)$ は画像 $I_i (1 \leq i \leq 1,000)$ の COREL 画像カテゴリを表し、クエリ画像 I_i に対して、 $r(i, j)$ は画像 I_j の検索結果中の順位 ($=1, \dots, 1000$) である。クエリ画像 I_i に対する適合率 $p(i)$ を以下のように算出する。

$$p(i) = \frac{1}{100} \sum_{1 < j < 1,000, r(i, j) < 100, ID(j) = ID(i)} 1, \quad (16)$$

これは検索結果の上位 100 位における適合画像の百分率である。

そして、1000 枚の画像データベースにおいて行った 1000 回の画像検索実験の全平均検索適合率 p を

$$p = \frac{1}{1,000} \sum_{i=1}^{1,000} p(i) \quad (17)$$

のようにして算出し、これを検索性能の評価に用いる。

5.2 ビジュアルキーの適合率への影響

ビジュアルキーを用いず従来手法同様、クエリ画像と対象画像間の類似度を部分画像から抽出した特徴量間の類似度から算出する手法では、全平均検索適合率 p は 0.377 である。(図 4 中の 2 点鎖線参照)。また、クエリ表現をビジュアルキーに制限し、クエリとして与えられた画像の各部分画像に対応するビジュアルキーをクエリとした場合、全平均検索適合率 p は画像データベースから抽出するビジュアルキーの数を増やすことで増加していくことがわかる。(図 4 中—参照) これはクエリの表現能力の向上によるものであり、データベースから 8,962 の部分画像が抽出される中から 200 個の部分画像を用いるだけで全部分画像を用いる場合の $p = 0.377$ の 90% に達することも図からわかる。

そして、画像データベースをビジュアルキーで索引付けする提案システムの全平均検索適合率 p を図中、 \oplus で示す。図より、データベースから抽出するビジュアルキーの数が 80 よりも小さい間は、ビジュアルキーの数が増えることで、上述のものより検索性能向上率が高いことがわかる。同様に、ビジュアルキーの数が 80 を超えた後は、検索性能が低下していることもわかる。検索性能の低下は、ビジュアルキーの数を増やすことでクエリに対してマッチする画像数自体が減少するためである。(式 (14), (10), and (11) 参照) ビジュアルキーが 80 前後の時に検索性能がピークを迎え、全部分画像を用いる従来型の手法の検索性能の 90% を達成することから、このデータベースに対する最適なビジュアルキーの数を 80 とする。この 80 という数は、1 枚の画像が約 9 つの部分画像に分割され、画像カテゴリが 10 あるということからも適切な値であると考えている。

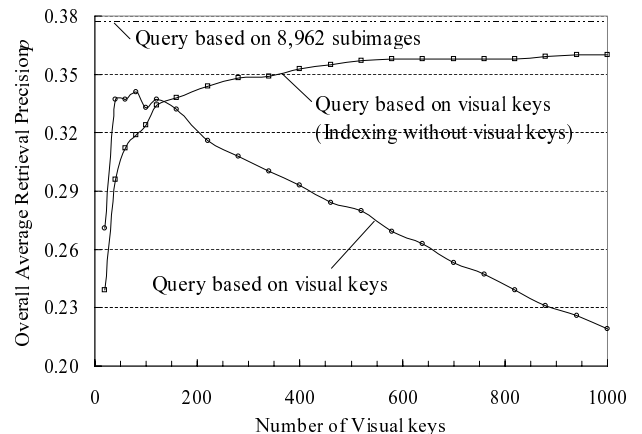


Figure 4: データベースから抽出するビジュアルキーの数と全平均検索適合率

5.3 検索速度

この章の始めで説明した環境において、提案システムにおける1回の検索に必要な時間は4ms以下である。索引付けが容易ではない従来のCBIRシステムでは画像から抽出した特徴ベクトル間の類似度を計算して検索結果を返す。そこで同様に提案システムにおいてインデックスを用いず特徴量間の類似度を算出して結果を出力すると、1回の検索に200ms以上かかる。このことから、クエリに画像データを用いるCBIRシステムであるにも関わらず、キーワードをクエリに用いる検索システムと同様、検索効率が良いことがわかる。

6 おわりに

テキストデータを考慮しない段階ではあるが、ビジュアルキーという画像検索における構成要素を提案し、定量的な検索性能実験を行った。そこから、全画像から抽出される8,962の部分画像から80程度の部分画像を用いることで、90%の検索性能とキーワード検索に類する検索効率を得ることが可能であることがわかる。

今後は、被験者実験によって定性的な評価を得、実際のユーザモデルとシステムモデルとのセマンティックギャップの評価を行う。また、テキストデータを考慮し画像内容解釈の補助をすることでそのギャップをより縮めた画像検索エージェントの研究を行う。

References

- [1] K. Sakuma et al.: "A Proposal of Visual Keywords in Image Retrieval," 30th Intelligent System Symposium, 2003.
- [2] M.Stricker and M.Orengo: "Similarity of color images Storage and Retrieval for Image and Video Databases III", Proc. SPIE 2420, pp.381-392, 1995.
- [3] J. Laaksonen et al.: "Analyzing Low-Level Visual Features Using Content-Based Image Retrieval," Proc. ICONIP 2000, Nov. 2000.
- [4] S. Brandt et al.: "Statistical Shape Features in Content-Based Image Retrieval," Proc. ICPR2000, Sep. 2000.
- [5] M. Flickner et al.: "Query by Image and Video Content: The QBIC System," IEEE computer, vol. 28, no. 9, 1995.
- [6] C. Carson et al.: "Blobworld: A System for Region-Based Image Indexing and Retrieval," Proc. Visual Information Systems, pp. 509-516, June 1999.
- [7] J.Z. Wang et al.: "SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries," IEEE Trans. Pattern Analysis and machine intelligence, vol. 23, no. 9, sep. 2001.
- [8] A. Gupta and R. Jain: "Visual Information Retrieval," Comm. ACM, vol. 40, no. 5, pp. 70-79, May 1997
- [9] S. Mukherjea et al.: "AMORE: A World Wide Web Image Retrieval Engine," Proc. World Wide Web, vol. 2, no. 3, pp. 115-132, 1999.
- [10] A. Natsev et al.: "WALRUS: A Similarity Retrieval Algorithm for Image Databases," SIGMOD Record, vol. 28, no. 2, pp. 395-406, 1999.
- [11] A. Pentland et al.: "Photobook: Tools for Content-Based Manipulation of Image Databases," Proc. SPIE, vol. 2185, pp. 34-47, Feb. 1994.
- [12] R.W. Picard and T. Kabir: "Finding Similar Patterns in Large Image Databases," Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing, vol. 5, pp. 161-164, 1993.
- [13] T.P. Minka and R.W. Picard: "Interactive Learning Using a Society of Models," Pattern Recognition, vol. 30, no. 3, p. 565, 1997.
- [14] J.R. Smith and S.-F. Chang, "VisualSEEk: A Fully Automated Content-Based Image Query System," Proc. ACM Multimedia, pp.87-98, Nov. 1996.
- [15] S. Stevens, et al.: "Informedia: Improving Access to Digital Video," Interactions, vol. 1, no. 4, pp. 67-71, 1994.
- [16] W.Y. Ma and B. Manjunath: "NeTra: A Toolbox for Navigating Large Image Databases," Proc. IEEE Int'l Conf. Image Processign, pp. 568-571, 1997.
- [17] Y. Rubner et al.: "The Earth Mover's Distance, Multi-Dimensional Scaling, and Color-Based Image Retrieval," Proc. DARPA Image Understanding Workshop, pp. 661-668, May 1997.

- [18] J.Z. Wang et al.: "Content-Based Image Indexing and Searching Using Daubechies' Wavelets," *Int'l J. Digital Libraries*, vol. 1, no. 4, pp. 311-328, 1998.
- [19] J. Shi and J. Malik: "Normalized Cuts and Image Segmentation," *Proc. Computre Vision and Pattern Recognition*, pp. 731-737, June 1997.
- [20] J.Z. Wang et al.: "Unsupervised Multiresolution Segmentation for Images with Low Depth of Field," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 1, pp. 85-91, jan. 2001.
- [21] K. Karu et al.: "Is There any Texture in the Image?" *Pattern Recognition*, vol. 29, pp. 1437-1446, 1996.
- [22] C. Carson et al.: "Term Weighting Approaches in Automatic Text Retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513-523, 1998.
- [23] <http://elib.cs.berkeley.edu/photos/blobworld/>
- [24] W.B. Frakes and R. Basza-Yates: "Information Retrieval: Data Structures and Algorithms," PrenticeHall, New Jersey, 1992.