

画像クラスタリングによる適合性フィードバックを利用した Web 画像検索システム (続報)

市川 哲彦[†] 佐々木英文^{††} 和田 忠之^{††}

[†] 山口大学メディア基盤センター 〒755-8505 山口県宇部市南小串 1-1-1

^{††} 山口大学工学部知能情報システム工学科 〒755-8611 山口県宇部市常盤台 2-16-1

E-mail: jichikay@yamaguchi-u.ac.jp

あらまし Web 画像検索では, HTML ファイル中に含まれる画像を, その周辺に出現する単語をキーとして検索を行う方法が一般的である. しかしながら, 必ずしも単語が画像特徴を良く表しているわけではないことから, 検索精度は必ずしも良くない. また, 検索結果の画像やそれらを参照している元文書の内容にもとづいて問合せ修正をする作業はすべてユーザに任されているのであるが, この過程での画像認識のユーザ負荷や, 問合せ修正の方針を立てるためのページ閲覧の負荷があり, 通常の Web ページ検索に比べて修正にかかるユーザ負担が大きいと言える. そこで本研究では, 検索結果に含まれる画像をクラスタリングし, その中から正事例を選択する適合性フィードバック処理によって画像認識負荷を軽減し, また, 選択されたクラスタに含まれる画像を参照している Web ページから問合せ拡張の候補を抽出してユーザに選択させることで問合せ修正の負担を軽減するシステムを開発した. 基本的な考え方を実装したプロトタイプについては既に [1] にて報告してあるが, 個別モジュールの実装の見直しを行い, また複数のサーチエンジンを使ったテストも進めているので, 本論文では続報として改めて報告を行う.

キーワード Web 画像検索, 画像クラスタリング, 適合性フィードバック, 文書ベクトル抽出

A Web image retrieval system using relevance feedback with image clustering (follow-up)

Yoshihiko ICHIKAWA[†], Hidefumi SASAKI^{††}, and Tadayuki WADA^{††}

[†] Media & Information Technology Center Yamaguchi University Minamikogushi 1-1-1, Ube, Yamaguchi, 755-8505 Japan

^{††} Department of Computer Science and System Engineering, Faculty of Engineering, Yamaguchi University Tokiwadai 2-16-1, Ube, Yamaguchi 755-8611 Japan

E-mail: jichikay@yamaguchi-u.ac.jp

Abstract Web image retrieval is performed based on the terms appearing near images in HTML files. However, the relationship between the terms and the images are not so accurate, the precision tends to be low. In order to improve the performance, relevance feedback is one of the promising approach. The user's burden, however, is the inherent defect of the approach. Moreover, the user must browse several pages in order to modify the query to improve performance, and so the burden to do it is another inherent problem. So, this research tries to lessen the burden by using relevance feedback based on the clusters of the retrieved images and to show the query terms extracted from the relevant Web pages referring to the images in the cluster. The initial prototype has been reported in [1]. This paper is a follow-up to the paper and describes improvements in the clustering method and the candidate selection mechanism from the selected positive cases, and explains the preliminary effectiveness study of the tool with different search engines.

Key words Web image retrieval, image clustering, relevance feedback, document vector extraction

1. はじめに

近年は Web ページ検索エンジンが普及しているが、加えて画像検索機能も一般化してきている [2]~[6]。これらのエンジンでは、HTML ファイル中で参照されている画像を、そのファイル中の単語で特徴付け、この特徴を用いて画像を検索する手法が取られている。これにより、画像の内容に基づく類似検索 [7] で問題になるページゼロ問題 (page zero problem) [8]、すなわち、類似検索においては初期画像をうまく選択しないと正しい検索結果が得られないという問題、を解消している。これらのシステムでは、テキストベースの特徴空間を用いた画像検索機能 [2], [3] や、その検索結果から引き続き内容に基づく類似画像検索 [5] を行う機能を提供している。

通常のキーワードベースの検索であっても、初期問合せで満足の行く結果が得られるわけではないため、試行錯誤的な問合せ修正のプロセスが必要とされる。画像検索の場合は、画像が含まれる HTML 文書が必ずしも詳しくかつ正確なキャプション付けをしているとは限らないことから、通常よりもこのプロセスは複雑になる。すなわち、利用者は検索された画像を認識してそれが自分が意図したものであるかを判定する処理と、検索結果を参照している Web ページを閲覧し、どのような単語がそこで使われていて、どのように問合せ修正をすれば所望の画像にたどり着けそうかを立案する処理を自身が行わなくてはならない。

効果的な性能改善の手法としては適合性フィードバック (relevance feedback) が知られている [9]。これは検索結果に対して利用者が正しい結果と間違っている結果の分類を行い、正しいと判断されたもの (正事例) のみが検索されるように問合せを修正する手法である。

本研究ではこの考えを Web 画像検索に適用した。検索結果の画像群から利用者が正事例や不事例を判定すると、システムはその結果に基づいて、画像群を参照している Web ページ群を分析し、問合せ修正を自動的あるいは半自動的に行う。ただし、単純に画像一覧を提示して正事例や負事例を選択させてしまうとユーザの画像認識の負荷が大きくなるという問題がある。

適合性フィードバックにおける利用者負担を軽減する方法としては、検索結果の文書集合のクラスタリングを行い、各クラスタの代表にだけ真偽の判定をする方法が知られている [10]。そこで本研究では、検索結果に対して画像特徴を用いたクラスタリングを行い、その結果をユーザに提示して判定をさせ、さらにその結果を問合せ拡張に利用するという手法を採用している。これにより、テキスト特徴を用いた Web 画像の試行錯誤的な検索プロセスにおいて、従来利用者自身が行っていた画像認識と問合せ修正処理のタスクを、画像のクラスタリングと候補単語抽出処理によって支援することが可能となる。

クラスタリングにおいてはクラスタ数、最大半径、最大許容誤差など、さまざまなパラメータを含める事になる [11]。これらのパラメータを利用者自身に選択させる方法も考えられるが、それでは、利用者負担を減らす為の処理が逆に自由度を生み、それが新たな試行錯誤プロセスという利用者負担につながってしまう。そこで、いくつかの予備実験に基づき、適切なクラスタリングを推測する手法を採用している。

HTML ファイル中の単語は対象画像との関係性を重みとして計算し、重みの大きいものから順に問合せ拡張候補として利用者に提示される。重みは画像からの距離に応じて決まる周辺

スコア、画像との親子関係 / 兄弟関係で決まる構文スコア、どの HTML 要素に含まれるかで決まるタグスコア [12] を利用して求めている。これが、単語頻度 (term frequency) に相当するファクタとなる。

Web 画像の文書ベクトルに基づく検索ではなく、画像の類似検索にたいして画像クラスタリングと適合性フィードバックを用いた研究として [13] があるが、クラスタ数は SOM のノード数というパラメータで固定されている点と、画像の特徴ベクトルのみを用いており、Web 画像検索のようにキーワードリストが問合せとなったときの適合性フィードバックの枠組みを示していない点が異なる。もちろん、本論文で提案するシステムのサブシステムとして利用する可能性はある。Web 画像検索への適用例としては [14] があるが、画像特徴はシステム利用者が真陽性と偽陽性とに検索結果を分類する際に (システム利用者自身により) 利用されるだけであって、システム側は画像特徴を利用していない。また、性能向上のために多数の画像を選択する方法をチェックすると、ユーザの負担が増すという問題もある。

基本的な考え方を実装したプロトタイプについては既に [1] にて報告してあるが、個別モジュールの実装の見直しを行い、また複数のサーチエンジンを使ったテストも行っているため、本論文では続報として改めて報告を行う。

適合性フィードバックの目的は少ない操作回数で短時間に所望のデータを見つけることにある。画像クラスタリングは、正事例の選択における操作と認識負荷を軽減するために行っている。また、クラスタリングされたデータを提示する際には縦方向のリスタリングをしているが、隣接しているクラスタ同士が似ているように配置し、余計なスクロールと画像識別をしないで済むようにしている。候補単語の抽出をしない場合は、利用者自身が Web ページを開き、問い合わせ単語以外にどのような単語が現れるのかを判断しなくてはならないが、自動抽出をすることで、ページを開き閲覧し候補を選び出し記憶するという一連の操作による負担を軽減することができる。

このことから、本システムは情報可視化の役割である思考コプロセッサ (cognitive coprocessor) [15] と同じ役割を Web 画像検索において果たしているものと考えられる。要素技術自体の新規性を追求するよりも、むしろこの役割に照らし合わせた時に機能するかどうかに主眼を置いて設計をしている。従って、出来るかぎり利用者選択パラメータを減らす事、また、処理時間は応答速度を悪化させない程度に軽いことに配慮している。

以下の本論文の構成は次の通りである。まず第 2 節では、ユーザとシステムとのやり取りがどのように進むのかという点と、システムのアーキテクチャ、すなわち、システムを構成するモジュールとそれらの機能および相互関係について説明する。次に第 3 節では、クラスタリングで利用される画像特徴および非類似度計算方法について説明し、続く、第 4 節では、クラスタリングアルゴリズムについて述べる。第 5 節で適合性フィードバックについて述べたのち、第 6 節で実験結果について報告する。最後にまとめと今後の課題について述べる。

2. システムの概要

本システムの構成要素とユーザとのやり取りを図 1 に示す。システムは主に PHP を利用して実装されているので、利用者入力に対しての処理と次画面提示というまとまりを意識して図

イメージ検索

検索したい画像に関する単語を入力してください。

インバラ

図 2 システム利用例: 問合せ入力画面

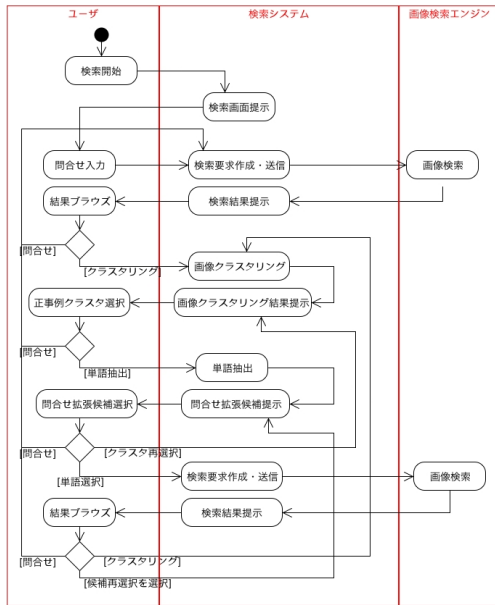


図 1 システム動作を表すアクティビティ図

が作成されている。

システムはまず、利用者からの問合せを受け取り、それを Web 画像検索のエンジンに送り、結果を利用者に提示する。本システムはメディアータとして動作するので、検索エンジンとしては Web サービス ([16] 等) の形で提供される検索エンジン機能を利用することを想定している。以下の記述では、特に断らない限り、Yahoo!Japan の提供する REST 型の画像検索サービスの利用を前提としている。

通常の検索システムでは、この結果を受けた後、候補画像の認識、画像の元ファイルの参照、問合せ修正方針の立案というプロセスをユーザ自身が行い、次の検索を行う。このサイクルは所望の画像が見つかるまで試行錯誤的に実行される。

本システムの目的はこの過程を適合性フィードバックによって支援することである。ユーザは検索結果のクラスタリング処理を要求することでフィードバックのステップを開始する。システムはこれを受けると、問合せ結果の画像群をクラスタリングする処理に入り、結果を利用者に提示する。これは、そのまま事例選択の画面となっているので、ユーザがその中から適切と考えるクラスタを正事例として選択する。この結果を受けると、システムはそのクラスタ中の画像群と対応付けられている HTML 文書群をダウンロードして画像周辺の単語を抽出し、問合せ拡張の候補単語を利用者に提示する。通常の適合性フィードバックでは問合せの自動修正までシステムが行うが、画像検索の場合にそこまで行うのは難しいという点と、どのような単語で問合せ拡張をするかを利用者に任せる事によって、システムに知識発見的な側面を持たせられる事から、候補単語を重みを付けて提示することとした。

図 2 に「インバラ」を検索語とする例を考える。インバラは動物の名前であると同時に車種名でもあるため、単純にこの単語を検索語として画像検索を行うと、双方の画像が含まれてしまう (図 3)^(注1)。ここでクラスタリングを指示すると図 4 のよ

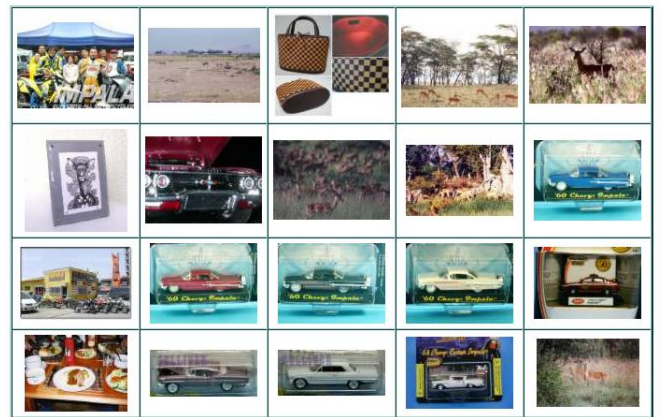


図 3 システム利用例: 画像検索結果提示画面 (参照画像の URL は付録を参照)



図 4 システム利用例: クラスタリング結果提示およびクラスタ選択画面 (実際の画面では順に表示される。参照画像の URL は付録を参照)

(注1): サムネイル画像は、元の Web ページへのアンカーになっているので、通

選ばれたクラスタ群から頻出単語上位40件を表示

初期検索単語を使わない場合はチェックをはずしてください

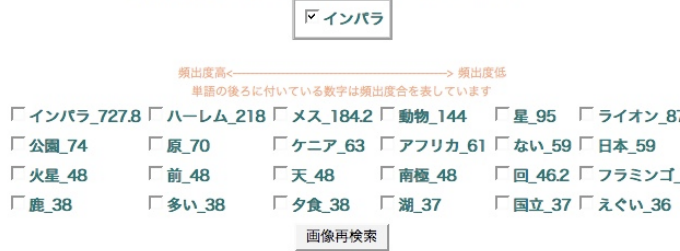


図 5 システム利用例: 問合せ拡張候補提示選択画面。提示される単語は次の通り (単語とその重みを交互に示す): インバラ 727.8 ハーレム 218 メス 184.2 動物 144 星 95 ライオン 87.4 草 80 カバ 79 人 78.6 ハット 76 公園 74 原 70 ケニア 63 アフリカ 61 ない 59 日本 59 ハイラックス 58.2 オランダ 54.6 大統領 54.6 ペリカン 52 火星 48 前 48 天 48 南極 48 回 46.2 フラミンゴ 45 観光 44 ヒョウ 43 いい 41 写真 41 鹿 38 多い 38 夕食 38 湖 37 国立 37 えぐい 36 パック 36 極楽 36 明るい 36 サファリ 35

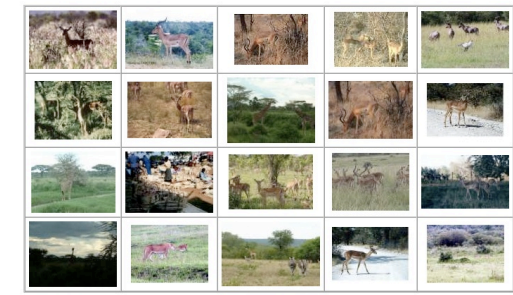


図 6 システム利用例: 再検索結果画面 (参照画像の URL は付録を参照)

うな結果が得られる。各クラスタの代表画像が最も左に表示され、その右側にクラスタの他の要素が並んでいる。画像の上に表示される数値は、代表画像との非類似度である。この段階で、概ね動物と自動車に分ける事ができているかが分かる。(注2)

クラスタ 3, 4 を正事例として選択した結果が図 5 である。現在負事例は選択できない。クラスタ中に含まれる画像の元のファイル群に対して、後述の単語抽出処理を行い、それらを重みの降順で並べたものがユーザに提示されている。重みについては後述する。ユーザはこの中から問合せ拡張に利用する単語を選択し、再度の画像検索を行う。ここでは「動物」を追加したとする。結果は図 6 のようになり、概ね動物のインバラの画像が検索されることが分かる。検索そのものはあくまで単語ベースのものであって画像認識などを使っているわけではないため、検索結果の中にはインバラではない画像も含まれている。

3. 画像間の非類似度計算手法

通常、色情報だけによる類似画像検索では、画像中に何が写っているかという情報が利用されないため、必ずしも良い結

果がでない。しかしながら、キーワードベースの Web 画像検索の後処理で画像の類似性によるクラスタリングをするという位置づけで考えた場合、何が画像中に含まれるかという情報は最初のキーワードベースの Web 画像検索の段階である程度与えられており、従って、問合せ結果には概ね所望の物が含まれていると考えることができる。このことから、画像の雰囲気や、背景等の画像が撮影されたシチュエーションの類似性を与える色特徴がより適切と考え、これを用いることとした。本研究では文献 [17] の方法を採用している。

この手法では画像の特徴量は次のように計算される:

(1) まず画像のカラーモデルを HSV に変換する。

(2) H, S, V の各チャンネル毎に平均, 分散, 歪みを計算する:

$$E_i = \frac{1}{N} \sum_{j=1}^N p_{ij}$$

$$\sigma_i = \left(\frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^2 \right)^{\frac{1}{2}}$$

$$s_i = \left(\frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^3 \right)^{\frac{1}{3}}$$

ここで N はピクセル数, p_{ij} はピクセル j の i チャンネルの値, E_i, σ_i, s_i はそれぞれチャンネル i の平均, 分散, 歪みである。

(3) 各チャンネルごとに得られた 3 つの値から 9 次元ベクトルを構成し、これを画像の特徴量とする。

二つの画像 H, I の平均, 分散, 歪みをそれぞれ $E_i, \sigma_i, s_i, F_i, \zeta_i, t_i$ ($i=H, S, V$) とすると、これらの画像間の非類似度は次式で定義される:

$$d(H, I) = \sum_{i=H, S, V} w_{i1} |E_i - F_i| + w_{i2} |\sigma_i - \zeta_i| + w_{i3} |s_i - t_i|$$

ここで、 w_{ij} ($i=H, S, V; j=1, 2, 3$) は重みである。文献 [17] では 3 種類の重みが試されている。(注3)

$$W_1 = \begin{pmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \\ 1 & 1 & 1 \end{pmatrix} \quad W_2 = \begin{pmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \\ 3 & 3 & 3 \end{pmatrix}$$

$$W_3 = \begin{pmatrix} 1 & 2 & 2 \\ 2 & 4 & 4 \\ 1 & 2 & 2 \end{pmatrix}$$

本システムでは、 W_3 を採用している。これは、予備テストの結果、彩度に重みを置く方が、より印象の似た画像により高い類似度を示す傾向が見られたためである。

4. 画像クラスタリング

ここではまず基本となるクラスタリングアルゴリズムと性質を表す簡単な実験結果について述べ、続いてクラスタリング数を動的に決定する処理について説明する。

4.1 基本となるクラスタリング処理

基本クラスタリング処理としては k-平均法 [11] を利用した。

常の Web 画像検索のインターフェースと同様の使い方も可能である。
(注2): 既報の [1] と異なり生成するクラスタ数は動的に決定される。

(注3): 文献 [17] の表 1 に示されたものの転置になっている点に注意。

サンプルデータの個数を n , 生成するクラスタの数を c , i 番目のクラスタを ω_i , ω_i の重心ベクトルを μ_i とすると, このアルゴリズムは次のようになる:

- (1) μ_i ($i=1..c$) を初期化する .
- (2) **do**
 - サンプルデータを μ_i との距離が最も近いクラスタ ω_i に分類する;
 - μ_i ($i=1..c$) を再計算する
- until** μ_i ($i=1..c$) に変化が無くなる

(3) **return** ω_i ($i=1..c$)

初期化は次のアルゴリズム [11] によって行っている:

- (1) サンプル点の重心を求め, これに最も近いサンプル点を μ_1 とする .
- (2) **for** $k=2..c$ **do**
 - $j := \arg \max_{i=1}^n \min_{l=1}^{k-1} d(p_i, \mu_l)$
 - $\mu_k := p_j$

ここで p_i ($i=1..n$) はサンプルデータである .

4.2 クラスタ数の動的な選択

先の実験では, 7 セットの画像をマージして得られるデータセットを 7 個のクラスタに分割した時に, 正しく分割できるのを見たのであるが, 実際にはいくつのクラスタに分割するのが適切かをあらかじめ固定しておく事は難しい. これは, クラスタリング対象の画像が多岐にわたる事と, そもそも, クラスタリング結果の妥当性は, 利用者の利用目的に照らし合わせて主観的に決定されるからである .

このことは, 利用者にクラスタ数を選択させる選択肢を示唆するものではあるが, クラスタリング処理の導入は, 検索結果を認識する負荷と, 正解画像を選択する操作回数を減らす事が目的であったため, 本システムが提供するサービスにおいては新たな利用者パラメータの導入は必ずしも望ましくない .

探索的なクラスタリングを行う手法として, 階層的なクラスタリングを行い, その中でどのレベル^(注4) が適切かを選択する手法がある [11]. レベルを対話的に選択するようにすることも可能であるが, 先の議論と同様に, 新規に対話パラメータを導入する事になり, 利用者の負荷の軽減にはつながらないという問題がある .

そこで我々は, 次のようにアルゴリズムを修正して対処する事とした. まず, k -平均法によって得られたクラスタの中で, もっとも直径^(注5) が大きいものを考える. これはクラスタリングの粒度を表していると考えられるので, これを g_k と書く事にする. 適切な g_k はやはり画像によって異なってくるので, これを自動的に決定することは難しいと考えられる. しかし, g_{k-1} と g_k の差^(注6) がある一定以上になった場合には, 粒度の変化が大きく, クラスタリングの性質が大きく変わったと看做することができるので, この閾値として適切なシステムデフォルト値 δ_{max} を与え,

$$\arg \max_{k \in N} (g_{k-1} - g_k \geq \delta_{max})$$

として, 自動的に適切なクラスタ数を選択することができれば,

(注4): 最も高いレベルでは, 全てが一つのクラスタになり, 最も低いレベルでは全ての要素がそれだけからなるクラスタになる .

(注5): クラス内の要素間非類似度の最大値 .

(注6): 相対差, 絶対差の双方が考えられるが, ここでは絶対差を考えている .

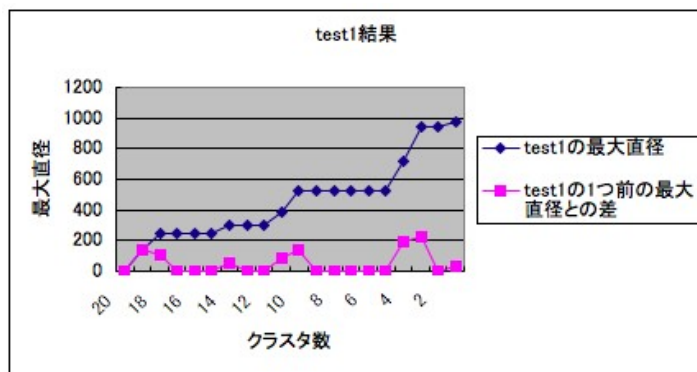


図 7 許容最大直径を変化させた時の最小クラスタ数の変化.「海」を問合せとして Yahoo! 画像検索の結果を実験対象としている .

これを利用することが考えられる .

この考えに基づいて予備実験を行った . 多様な画像が入りやすい「海」「絶景」「鮮やかな」「動物」「花見」という 5 つのキーワードでの Yahoo! 画像検索の結果を実験対象としている . まず, 各問合せ結果について人手でクラスタリングを行い, 正解集合を決定した . 次に, k の変化に対して g_k がどのように変化するかを調べた . 図 7 に「海」を検索語として用いた時の変化を示す . このデータセットの正解クラスタ数は 11 なので, 正解周辺で最大直径に変化があることが分かる . ただし, クラスタ数がより多い範囲においても, また, 少ない範囲においても同様な変化が現れている . そこで, 最大直径の範囲を限定することで正解あるいはその周辺のクラスタリングを見つけるようにアルゴリズムを構成することを考えた . このアルゴリズムは, δ_{max} と g_{min} をチューニングパラメータとして次のように構成されている:

- (s1) $k :=$ “画像数” とする;
- (s2) $k = 0$ ならば 1 を正解として終了;
- (s3) g_k と g_{k-1} を求める;
- (s4) $g_{k-1} - g_k \geq \delta_{max}$ で無ければ $k := k - 1$ として (s2) へ;
- (s5) $\max(g_{k-1}, g_k) \geq g_{min}$ でなければ $k := k - 1$ として (s2) へ;
- (s6) $g_k \geq g_{min}$ ならば k を, そうでなければ $k - 1$ を界として終了 .

5 つのテストによる実験から, $g_{min} = 400$, $\delta_{max} = 20$ とした .

4.3 クラスタ提示順序

ここでクラスタの提示順についても簡単に説明をしておく . クラスタを利用者に提示する時に適切なりスティングを行う事で事例チェックとを途中で打ち切ることができるため, これにより利用者負担を軽減することができる . そこで次のようにした:

- (s1) クラスタ群を S とす;
- (s2) S 中の最大直径のクラスタ C を提示し, $S := S - C$ とする;
- (s2) **while** $S \neq \emptyset$ **do**
- (s3) S から最後に提示したクラスタと最も重心に近いクラスタ C を求める;
- (s4) C を提示し, $S := S - C$ とする;
- (S5) **done**

5. 適合性フィードバック処理

上述のクラスタリングの処理の結果を利用者に提示して適切なものを選択させることで適合性フィードバックの処理を行う。

試作したシステムは次のような手順で処理を行う:

(1) 利用者から提示された問合せを用いて上述のサイトにアクセスし、上位 N 件分のデータをダウンロードする。画像としては検索エンジンの提供するサムネイルを、また Web ページとしてはオリジナルのものをダウンロードする;

(2) 検索結果の画像を前節で述べたアルゴリズムを用いてクラスタリングし結果をユーザに提示する;

(3) 選択されたクラスタに含まれる画像を参照する Web ページを読み取り、キーワード抽出を行う。(抽出アルゴリズムは後述する。)

応答時間を短縮するため、ファイル群のダウンロードは同時並行で行い、一定時間内にレスポンスが無い場合は、ダウンロードを中断し、分析対象から外している。

HTML ファイルの処理は DOM [18] を用い、PHP5 によって記述している。この前処理として HTML 形式から XML 整形形式に変換する必要があるため、OpenJade [19] と共に配布されている `osx` を使って行って文書の正規化を行っている^(注7)。構文エラーや文字コードの検出エラーも発生するが、そのような文書は処理対象から除外している。

画像の周辺テキストの抽出処理について説明する。まず XPath を用いて IMG 要素と A 要素を見つけ、その中から画像の参照位置を特定する^(注8)。特定された位置から DOM 木の構造に従って前後の要素を探索する。探索の対象となるのは、TEXT 要素、IMG 要素の ALT 属性、TABLE 要素の SUMMARY 属性である。周辺だけを考慮するので、抽出する個数には前後 ε 個という上限を設けている。抽出されたテキストには [12] の方式を参考にして、画像位置からのリーフレベルの抽出対象ノードを並べた距離 d から決まる周辺スコア ($= \varepsilon - d$)、画像位置との親子・兄弟関係で決まる構文スコア (s とする)、テキストがどのような要素に含まれているかで決まるタグスコア (t とする) を付与し、テキスト中の単語には

$$(s + \varepsilon - d) \times t$$

という重みを付与している。 ε は現在 10 としている。構文スコアは、対象画像の親を根とする部分木のノードに 10、それ以外で、親の親のノードを根とする部分木のノードに 9、というような単純な割り振り方をしている。タグスコアは、画像位置にあるテキストに 10、title に 10、h* に 5.5、b や strong に 4.3 その他に 1 という降り方をしている。これらのパラメータもチューニング対象である。

抽出されたテキストからの単語抽出処理は、茶釜 [20] を用いている。形態素解析の結果から、品詞が、名詞(記号以外)、形容詞(非自立語を省く)、形容動詞のもののみを取り出し、さらに、ストップワードを取り除くことで処理を行っている。各単語のスコアはそれが現れるテキストのスコアを足したものであ

(注7): HTML ファイルを直接読み込む機能もあるが、日本語コードの取扱いに不具合があるようである。

(注8): 2 段階に分けているのは、文字コード、HTML エンコード、相対パスの処理を行う必要があるため、単純に XPath の属性条件だけを使って特定することは難しいためである。

る。これが単語頻度 (term frequency) に相当する。通常は文書中の単語数で正規化するが、ここでは一定の範囲でしか探索をしないことから、特に正規化はしていない。

問合せ拡張は、上記の処理を正解として選択されたクラスタ中に含まれる HTML ファイルと画像のすべてに對して行い、その結果をマージすることで行っている。この時、単語の重みは単純に足し合わせている。

TF・IDF 法 [9] と同様の補正として、単語 w による単純検索のヒット数 h_w を調べ、それによって単語の重み tf_w を例えば

$$tf_w \cdot \frac{1}{\log h_w}$$

と補正することも検討したが、システムの応答時間が劣化すること、分析の結果を見る限りでは特にこの補正の必要は無いと判断し、実施していない。

6. 利用例と考察

開発したシステムを用いた画像検索例をいくつか説明し、適宜考察を加える。

まず、最初に触れた「インバラ」を検索語とする例をして本システムの特徴を再度説明する。既に述べた通り、インバラは動物の名前であると同時に自動車の名前でもあるため、単純にこの単語を検索語として画像検索を行うと双方の画像が含まれてしまう。ここではユーザが動物のインバラの画像を欲しているものと想定する。通常の画像検索システムでは、ユーザは、得られた画像の中で動物のインバラが写っているものを選択し、それらの元の Web ページをブラウズし、次にどのような問合せを構成したら良いのかを思考しなくてはならない。

本システムでは適合性フィードバックを用いるので、システムがこのプロセスを支援することになる。まず、すべての画像を調べる認識負荷を軽減するために、画像クラスタリングをした上で正事例を選択できる様にしている。この例では、クラスタリング処理で、概ね動物と自動車に分ける事ができるため、利用者はいずれの画像が欲しいのかを容易に指示する事ができる。先に示した例では、動物のインバラがクラスタの代表画像になっているクラスタ 3, 4 を選択している (図 4)。

これを受けシステムが元の Web ページをダウンロードして解析し、先に示したアルゴリズムを用いて問合せ拡張の候補を選び出す。結果は、図 5 に示すように重みの降順で提示される。ユーザは、動物のインバラの画像を欲している(この前提である)ので、「動物」を選択して問合せ拡張に用いる。結果は図 6 に示すような画像群であり、概ね動物のインバラの画像が得られていることが分る。ただし、問合せ拡張は、正事例として選択されたすべての画像に共通の単語によってなされているわけではないため、最終的な検索結果に、途中段階で正事例として選択された画像が含まれない可能性も高い点に注意が必要である。また、検索そのものは単語ベースのものであって画像特徴を使っている訳でないため、無関係な画像も検索される。

本システムの用途の一つとして、感性語彙による画像検索がある。つまり、形容詞や形容動詞を使った検索からスタートし、適合性フィードバックを繰り返しながら所望の印象を与える画像を得るツールでもある。既報の [1] では、文献 [21] に挙げられている感性語彙 110 個を利用して [6] を検索エンジンとして使ったときに、どのような検索ができるかを調べた結果の一部として、「幻想的な」をキーワードとした時の検索結果について



図 8 「幻想的な」による livedoor 画像検索をクラスタリングした結果の一部 (参照画像の URL は付録を参照)

述べた．同様に「鮮やかな」をキーワードにして花のクラスタを調べると「クンシラン」のような単語が上位に上がり，ここから，この花の写真の一覧にたどり着く事が出来る．この時の操作は，クラスタ選択，ポタクリック，単語選択，ポタクリックの4回であり，通常のインターフェースにおける，サムネイルのクリックとテキストの閲覧を複数回繰り返した後に，候補単語を入力してポタクリックという操作に比べて，少ない操作で検索が可能であることがわかる．

本システムはあくまでメディアータであってそれ自体は画像検索エンジンを持たない．現在は [6], [16], [22] に対応している．Blog ページの画像検索を与える [22] を使った時の本ツール利用例についてもここで説明をする．まずうまく機能する例として「幻想的な」からスタートするケースを考える．

上位 60 件を検索してクラスタリングすると，25 個のクラスタに分割された^(注9)．内訳は，6 個の要素を持つものが 2 個，5 個のものが 3 個，3 個のものが 4 個，2 個のものが 5 個，単独となったものが 11 個である．要素数が多いもののいくつかを

(注9): 先のインバラの例では 20 件にして検索をしている．

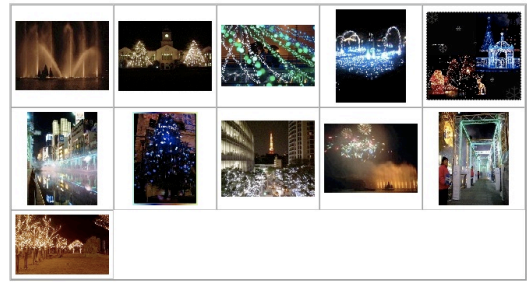


図 9 イルミネーションを拡張単語として検索した結果 (参照画像の URL は付録を参照)

図 8 に示す．印象の似た画像がクラスタに分類されていることが分かる．

ここからクラスタ 7 を選択して単語を抽出すると，上位 10 件は次のようになる: 画像 1383, クリスマス 159, 幻想 159, イルミネーション 85, パーク 72, フラワー 72, テーマ 66, トンネル 65, アート 61, 夜空 61．ここから例えばイルミネーションを検索すると結果は図 9 のようになり，選択したクラスタ 7 と同様の画像が選択されることが分かる^(注10)．

その他の印象語による検索も試みているが，通常の Web 画像検索に比して，期待したほど良い成果は得られていない．原因としてはいくつか考えられる: (1) 通常の Web 画像検索とは異なり，Blog 画像では画像周りのテキストが少なく，また，周辺テキストが必ずしも画像の説明になっているとは限らない; (2) blog 形式でオンラインショッピングのアイテムを説明をしたページが多く存在し，そのようなページでは美しい，繊細な，などの評価性 (evaluation) の観点で良い印象を与える語は商品説明に使われていて画像印象の説明になっていない．

逆に商品検索ツールとして位置づけることもできる．同じサイト内の商品写真は同様の条件で撮影されているケースがあり，その場合は，同じクラスタ内にそれらの商品が入るので，特定サイトあるいはページを素早く見つける事が出来る．さらに，ブログの作者は，何かの商品を購入したという場合に，詳細は省略して簡略な名称だけを入力し，画像を埋め込んだり他サイトの商品画像へのリンクを埋め込む事もある．そのため，本ツールを利用すると，同様な商品の画像を含むページを調べることができ，特定アイテムについての購買意欲，推薦，購入後の感想などといったものを調べる事が出来る．通常ブログのコミュニティ構造分析 [23] では，トラックバックやリンクといった相互参照，単語共起，クラスタ分類などのページ間関係が利用されるが，このような類似画像参照の関係からもページ間関係を示しているものにとらえる事ができると考えている．

7. まとめと今後の課題

本論文ではキーワードベースの Web 画像検索において適合性フィードバックを行うシステムを提案した．正事例の選択の際に画像クラスタリングを行うため，利用者の認識負担および事例選択負担を低減することができる．本システムでは，選択された画像群の元の Web ページを解析して画像の周辺単語を抽出し，それらを問合せ拡張の候補として利用者に提示する．

(注10): これが正解かどうかは，利用者の検索意図による．ここでは，とりあえず幻想的な印象を与える画像をいくつか見つけたいという意図があると仮定している．

本来の適合性フィードバックでは問合せ拡張はシステムが自動的に行うのであるが、Web 画像検索の場合は、単純に上位のものを追加すれば済むのでは無いため、候補を利用者に提示し、その中から単語を選択する方式を採用している。

文献 [1] では基本的な考え方と知識発見ツールとしての側面について報告したが、本稿ではクラスタリング処理や単語抽出処理の見直しについて、バックエンドの検索システムをブログ画像検索とした場合についての報告をした。印象語による検索ではうまく行くケースとそうでないケースが両極端になってしまったが、類似画像参照の関係から blog ページ間の関係をとらえることもできるのではと考えている。本件については更なる調査が必要である。

本システムは数多くパラメータを含むシステムとなっている。特に、画像クラスタリングに用いるパラメータをどのように決定するかという点と、HTML 文書からのテキスト抽出および単語の重み付けの処理が重要であるが、利用者に任せる事は、対話的システムの操作回数の増加を招いてしまうため、応答時間を損なわない範囲でどうこれを調整するかは引き続き検討する必要がある。また、一般利用者を被験者とする性能評価実験や定量的な性能評価についても引き続き検討したい

文 献

- [1] 市川, 菅, 立野, 菊政: “画像類似度でのクラスタリングによる適合性フィードバックを利用した web 画像検索”, 第 17 回データ工学ワークショップ (DEWS2006), 宜野湾, pp. 5B-02 (2006).
- [2] Google Inc.: “Google image search engine”, <http://images.google.com>.
- [3] AltaVista Company: “Altavista image search”, <http://www.altavista.com/image/>.
- [4] Lycos Inc.: “Lycos multimedia search”, <http://search.lycos.com>.
- [5] NTT レゾナント: “Goo ラボマルチメディア検索実験”, <http://mmm.nttr.co.jp/>.
- [6] Yahoo!Japan: “Yahoo!検索”, <http://search.yahoo.co.jp>.
- [7] V. N. Gudivada and V. V. Raghavan: “Content-based image retrieval systems”, IEEE Computer, **28**, 9, pp. 18-23 (1995).
- [8] M. L. Cascia, S. Sethi and S. Sclaroff: “Combining textual and visual cues for content-based image retrieval on the world wide web”, Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries, pp. 24-28 (1998).
- [9] R. Baeza-Yates and B. Ribeiro-Neto: “Modern Information Retrieval”, Addison Wesley (1999).
- [10] 湯木野, 松下, 平澤: “クラスタに基づいた適合性フィードバック手法”, 第 4 回情報科学技術フォーラム (FIT2005) 講演論文集, pp. D-044 (2005).
- [11] R. O. Duda, P. E. Hart and D. G. Stork: “Pattern Classification/2ed”, Wiley (2001).
- [12] 杉尾, 竹野, 藤本, 萩原: “WWW に対するマルチメディアデータ検索エンジンの HTML 構文を活かしたスコア付け手法の提案”, 第 13 回データ工学ワークショップ (DEWS2002), 倉敷, pp. B4-8 (2002).
- [13] 中島伸介, 木下真一, 田中克己: “差異増幅型適合フィードバックに基づく画像データベース検索”, 電子情報通信学会論文誌 (データ工学特集号), **J87-D1**, 2, pp. 164-174 (2004).
- [14] 竹安, 獅々堀, 拓殖, 北: “WWW 画像検索システムにおける検索質問拡張に基づくフィードバック検索”, 第 4 回情報科学技術フォーラム (FIT2005) 講演論文集, pp. D-019 (2005).
- [15] S. K. Card, J. D. Mackinlay and B. Shneiderman: “Information Visualization”, chapter 1 (1999).
- [16] Yahoo!Japan: “Yahoo!デベロッパネットワーク”, <http://developer.yahoo.co.jp/>.
- [17] M. A. Stricker and M. Orengo: “Similarity of color images”,

Storage and Retrieval for Image and Video Databases (SPIE), pp. 381-392 (1995).

- [18] World Wide Web Consortium: “Document object model (dom)”, <http://www.w3c.org/DOM/>.
- [19] OpenJade Project: “Openjade distribution page”, <http://www.openjade.org/>.
- [20] 奈良先端科学技術大学院大学: “形態素解析システム茶筌”, <http://chasen.naist.jp/hiki/ChaSen/>.
- [21] 千々岩: “色彩学”, 福村出版 (1983).
- [22] livedoor Co., Ltd: “livedoor 検索画像 版”, <http://blogimage.livedoor.com/>.
- [23] 奥村: “ブログマイニング”, 情報処理学会研究報告 2006-DBS-139/2006-FI-83, pp. 33-44 (2006).

付 録

1. 図3, 図4で参照した画像の URL 一覧 (<http://>は省略)

ktm-impala.com/img/top/tes.jpg
www.hpmix.com/home/alm/images/img194.jpg
www.7yorku.com/goods_p/big/4964.jpg
kobe.cool.ne.jp/masao/kenya/inpara.jpg
www.saso.jp/image/inpara2.jpg
www.aqua-net.org/gyamando/miyoshi/mt02-j/img1042158417.jpeg
kihatsu.fc2web.com/QS/dscn0323.jpg
www.saso.jp/image/inpara.jpg
ws.3lrm.ne.jp/free21/02mod12/okabango/oka10.jpg
www.gulliver-inc.com/j/car/others/image/12211-blue.jpg
ktm-impala.com/img/shop/shop1.jpg
www.gulliver-inc.com/j/car/others/image/12211-r.jpg
www.gulliver-inc.com/j/car/others/image/12211-b.jpg
www.gulliver-inc.com/j/car/others/image/12211-w.jpg
www.gulliver-inc.com/j/car/others/image/97058.jpg
www.remus.dti.ne.jp/t-noro/2001africa/tourphoto/FH000019.jpg
www.gulliver-inc.com/j/car/others/image/86-3391.jpg
www.gulliver-inc.com/j/car/others/image/86-3393.jpg
www.gulliver-inc.com/j/car/others/image/86-3351.jpg
pomelo.cool.ne.jp/travel/kenya/33-4.jpg

2. 図8で参照した画像の URL 一覧 (<http://>は省略)

marimo125.img.jugem.jp/20061213.229608.jpg
userdisk.weby.biglobe.ne.jp/005/138/15/1/116703737115813254.jpg
userdisk.weby.biglobe.ne.jp/006/933/70/1/116705922415832551.jpg
userdisk.weby.biglobe.ne.jp/007/854/03/1/116705160615531147.jpg
userdisk.weby.biglobe.ne.jp/007/854/03/1/116705156715522528.jpg
userdisk.weby.biglobe.ne.jp/007/854/03/1/116705158815522528.jpg
userdisk.weby.biglobe.ne.jp/007/854/03/1/116705146915516324.jpg
userdisk.weby.biglobe.ne.jp/007/854/03/1/116705164615514429.jpg
image.blog.livedoor.jp/tmon1/imgs/c/d/cdbd9e83-s.jpg
www.k4.dion.ne.jp/pakupuku/LOVELOG_IMG/061223-1754_0001.jpg
4color.cocolog-nifty.com/fullcolor/images/061223d.jpg
userdisk.weby.biglobe.ne.jp/003/573/48/1/2006-12-23-007.jpg
www.k3.dion.ne.jp/kochan/LOVELOG_IMG/0612200952083R83s815B.JPG
blogimg.goo.ne.jp/user-image/4b/96/db4d142d2b1f0015d47749eb3b360d2.jpg
userdisk.weby.biglobe.ne.jp/005/138/15/1/116703735215731780.jpg
pds.exblog.jp/pds/1/200612/22/62/d0097362_0551065.jpg
pds.exblog.jp/pds/1/200612/22/62/d0097362_112658.jpg
4color.cocolog-nifty.com/fullcolor/images/061223c.2.jpg
pds.exblog.jp/pds/1/200612/20/88/b0093088_6324128.jpg
pds.exblog.jp/pds/1/200612/22/32/f0117332_8213922.jpg
ngo-hokkaido.img.jugem.jp/20061222.205567.jpg
flower-c.up.seesaa.net/image/10040008-4-thumbnai2.jpg
pds.exblog.jp/pds/1/200612/20/85/f0120185_1465017.jpg
image.blog.livedoor.jp/mayubi/imgs/1/2/12b6ac1.jpg

3. 図9で参照した画像の URL 一覧 (<http://>は省略)

miyabi-staff.up.seesaa.net/image/8FAAB18FAAC18FABB18FAAD38FAAA120(17)-thumbnai2.JPG
userdisk.weby.biglobe.ne.jp/002/011/45/1/DSC03062.jpg
pds.exblog.jp/pds/1/200611/20/37/e0051637_18153729.jpg
pds.exblog.jp/pds/1/200611/25/07/d0070007_22234812.jpg
felice-vita.img.jugem.jp/20061214.267945.jpg
4color.cocolog-nifty.com/fullcolor/images/061223c.2.jpg
userdisk.weby.biglobe.ne.jp/006/443/84/1/116511299915723630.jpg
blogimg.goo.ne.jp/user-image/57/07/9db9b3e3069ceb25c27808f0694c2aa.jpg
miyabi-staff.up.seesaa.net/image/8FAAB18FAAC18FABB18FAAD38FAAA120(29)-thumbnai2.JPG
4color.cocolog-nifty.com/fullcolor/images/061223d.jpg
illuminationinfo.up.seesaa.net/image/ehime2.jpg