

ISDL Photo

渡辺 章人

1 はじめに

近年, Flickr や Picasa などの画像アップロードサービスにより, 画像をサーバ上で管理する機会が増えた. これらは, 画像の保存だけではなく, 他者と画像を共有したり, 画像に対してタグ等のデータを付与することで, 画像データへの情報の付与を可能にしている.

本研究室では, 研究生活の記録の一環として, 発表会などのイベント時に写真の撮影を行っている. しかしながら, それらのほとんどがファイルサーバーにアップロードされた後, 利用されないまま放置されている. このような状況を改善すべく, 知的システムデザイン研究室内の写真の有効活用を目的とした研究室画像管理システム「ISDL Photo」を考案, 構築した.

本システムでは, 画像のアップロード機能を機軸とし, 画像の表示機能, 検索表示機能, 画像への付加情報の追加機能, カレンダー情報との連携機能等を有する. また, 本システムでは OpenCV を利用した顔認識機能を実装し, 画像の中で最も重要となる“顔情報”の記述をサポートしている.

本報告では, 「ISDL Photo」のシステム説明と共に, 画像に対する情報の付与を目的とした顔認証機能の検討とその実装について述べる.

2 ISDL Photo

2.1 システム構成

本システムは, 動作環境として, Debian GNU/Linux4.0, Ruby 1.8.5, MySQL 14.12, Ruby On Rails 1.2.3 を利用している. また, HTTP サーバは Ruby で構築された WEBrick を利用している.

2.2 機能

実装されている機能は以下の通りである.

- ユーザー管理機能
- 画像のアップロード
- 画像の表示機能
- 画像へのテキスト情報書き込み機能
- 画像の顔画像認識機能
- Exif 情報を利用したカレンダー機能
- お気に入り機能

上記の機能の中で, 「ISDL Photo」の利用方法及び構築したシステムの特徴的な機能である「登録画像の顔画像認識」, 「Exif 情報を利用したカレンダー機能」の詳細について説明する.

2.2.1 画像の表示機能

本システムのトップページでは, Fig. 1 のようにアップロードされた画像が一覧となって表示される. また,

Fig. 1 の上部の検索機能を利用することによって, キーワード検索も可能である. この検索機能では, 2.2.2 節で説明を行う“テキスト情報書き込み機能”によって追加された情報が検索の対象となる.



Fig.1 画像の一覧表示機能 (出展: 自作)

一覧に表示された画像には, リンクが張られており, それぞれの画像の詳細表示ページへ遷移することが可能である. Fig. 2 に詳細ページの表示を示す.



Fig.2 画像の詳細表示機能 (出展: 自作)

詳細表示ページでは, 次節で説明する“登録画像のテキスト情報書き込み機能”が利用可能となる.

2.2.2 テキスト情報書き込み機能

画像のアップロードを行うと, その画像に対してテキスト情報の書き込みを行うことが可能となる. 機能は, 画像に対するコメント書き込み機能と画像に対して視覚的な情報追加を行う矩形選択書き込み機能の 2 種類がある. 前者はブログなどに見られる簡単なコメント機能に, 一階層のスレッド機能を付与したものである. 後者は, 画像の中で情報を添えたい部分を矩形に囲み, その矩形に対して情報を書き込むものである. その様子を Fig. 6(a) に示す. また, この機能により情報が書き込まれた

矩形部分は、その情報と共にデータベースに登録され、以後、Fig. 6(b)のように表示される。入力した情報に対しては、書き直しや削除が可能となる。また、入力した情報は、画像検索の対象となる。

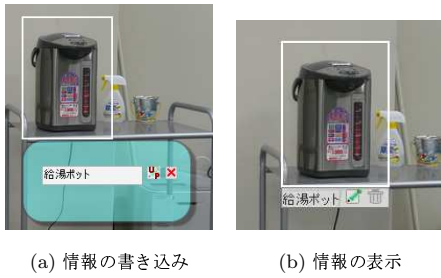


Fig.3 情報書き込み機能 (出展：自作)

2.2.3 顔画像認識機能

登録画像内の顔画像認識機能とは、画像アップロード時に対象画像内に含まれる“顔”を検知する機能である。この機能では、画像処理ライブラリ OpenCV を利用している。OpenCV を用いた顔画像認識については、第3章にて詳しく説明を行う。Fig. 4 に顔画像認識の結果を示す。

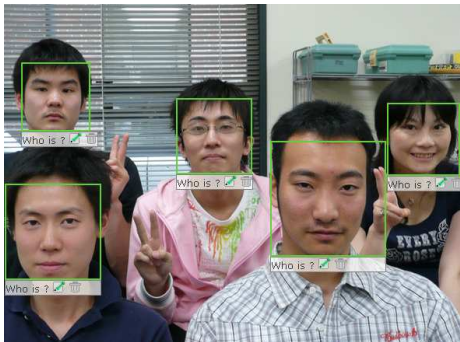


Fig.4 顔画像認識機能 (出展：自作)

この機能によって抽出された顔は、抽出と同時にデータベースに登録される。以後、前章で説明を行った「テキスト情報書き込み機能」と同様の振る舞いをする。

2.2.4 Exif 情報を利用したカレンダー機能

Exif(Exchangeable Image File Format) とは、1994年に富士フィルムが提唱したデジタルカメラ用の画像ファイルの規格であり、画像に撮影情報(Exif 情報)を付与することが出来る。現在では、ほぼ全てのデジタルカメラが Exif 形式で画像を保存するようになっている。Exif 情報には、大きさ、撮影日、カメラのモデル、種類等の情報が付与されている。本システムでは、アップロードされた画像の撮影日情報を利用することにより、カレンダー機能の実装を行った。実装したカレンダー機能の表示例を Fig. 5 に示す。

表示されるカレンダー内で表示される写真アイコンは、アップロードされた画像の中でその日に撮影されたものが存在することを示す。また、アイコンの横に記載され



Fig.5 Exif 情報に基づくカレンダーとの連携 (出展：自作)

ている数字は画像の枚数を示す。アイコンにマウスオーバーすると、カレンダーの下にその日に撮影された画像の一覧が表示される。

3 OpenCV

OpenCV(Intel Open Source Computer Vision Library) とは、米 Intel がオープンソースで公開しているコンピュータビジョンのためのライブラリである。画像処理やコンピュータビジョン必要となるアルゴリズムが、C 言語の関数や C++ のクラスで実装されている。¹⁾

3.1 OpenCV を用いた顔画像認識機能の実装

「ISDL Photo」では、OpenCV に実装されている“cvHaarDetectObjects”という関数を利用することにより顔画像認識を可能にしている。この関数は、高速物体検出を可能とする Haar-like 特徴 (Haar-like features) を応用することにより、物体の抽出を可能とする。Haar-Like 特徴量とは、矩形型のフィルター (Fig. ??(a),(b)) を用いて求めた隣接する画像中の矩形領域間の平均明度の差のことである。

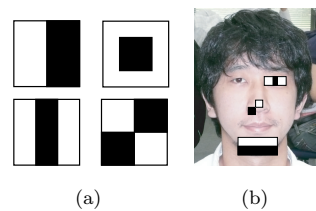


Fig.6 矩形型フィルターの例 (出展：参照²⁾)

これを用いて作成した弱識別器 (精度の低い識別器) に重みをつけ、組み合わせることによって、検出対象となる物体の検出を行っている。また、“cvHaarDetectObjects”を利用し、物体認識するには、その物体を検知するための識別器を作成する XML が必要となる。本システムでは、ライブラリと共に配布されている“haarcascade_frontalface_alt2.xml”という人間の顔画像を識別するための XML を利用した。

4 顔認証

顔認証とは人間の体の特徴を利用するバイオメトリクス認証の一種で、顔認識技術を用いて個人を認証するシステムを指す。顔認証は人の顔の画像データから特徴を抽出し、あらかじめ保存されたデータと照合することで個人の特定を行う。

この機能を本システムに組み込むことにより、画像をアップロードしただけでその画像に写っている被写体の情報を読み取れる。これにより、ユーザーに負担を掛けることの無い、画像への情報付与が可能となる。

本システムでは OpenCV を利用することによって、顔画像認識を行い、また、その認識について一定の精度があることを確認した。この機能を応用した顔認証機能の実装とその検討について説明する。

4.1 OpenCV を利用した顔認証の検討

OpenCV には、学習機能が備わっており、認識したい物体が写っている画像 (ポジティブ画像) と認識したい画像が写っていない画像 (ネガティブ画像) を用意することにより、任意の物体を抽出するために XML の作成が可能である。しかしながら、これを実現するには、ポジティブ画像 7000 枚、ネガティブ画像 3000 枚を用意する必要があると考えられる。そのため、この方法はサンプル数が比較的多い抽象的なもの、例えば“顔”といったものを検知するためには有効と思われるが、サンプル数の少ない特定のもの、例えば“A 君の顔”といったものを抽出するには有効ではない。従って、顔認証を実装するに当たり、他の手法を検討し、実装を行った。次節にて、そのアルゴリズムの説明を行う。

4.2 実装した顔認証機能のアルゴリズム

実装したアルゴリズムでは、顔認識の際に利用した“cvHaarDetectObjects”を用い、右目、左目、鼻、口の抽出を行う。抽出した右目、左目、鼻、口の中心点を顔の特徴点とし、これに基づくパラメータを作成、比較を行う。以下にアルゴリズムを示す。

1. アップロードされた画像から顔画像認識を行い、認識された顔画像を切り取る。
2. 切り取った画像より、右目、左目、鼻、口の抽出を行う。各部位の抽出には、それぞれに適した XML (Haar Cascades³⁾) にて配布) と、“cvHaarDetectObjects”を用い抽出を行う。各部位の抽出結果の例を Fig. 7 に示す。
3. 抽出した右目、左目、鼻、口の中心点を顔の特徴点とする。特徴点の抽出結果を Fig. 8 に示す。
4. 抽出した特徴点より、以下のパラメータの計算を行う。
 - 顔の横幅に対する目の距離 (Fig. 8-a) の比率
 - a に対する鼻から右目に掛けての距離 (Fig. 8-b) の比率
 - a に対する鼻から左目に掛けての距離 (Fig. 8-c) の比率

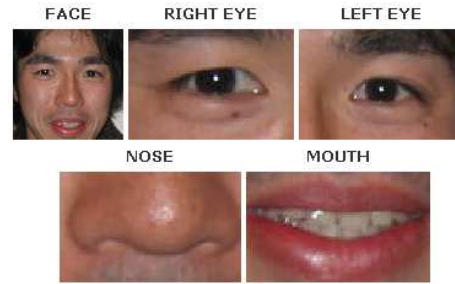


Fig.7 各部位の抽出結果例 (出展：自作)

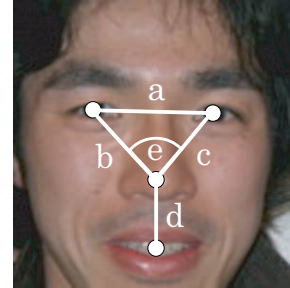


Fig.8 特徴点の抽出 (出展：自作)

- a に対する鼻と口の距離 (Fig. 8-d) の比率
 - b, c の二直線の間の角度 (Fig. 8-e)
5. 計算したパラメータを既に登録されている顔画像のパラメータと比較する。比較では、それぞれのパラメータについて誤差を計算し、昇順のリストを作成する。誤差が最も少ないものを重み 10、2 番目に少ないものを重み 9 とし、10 番目までのものに重みをつける。
 6. 全てのパラメータについて、同様の作業を行った後、重みの加算を行う。
 7. 加算した結果を顔認証の結果とし、降順に 10 番目までを候補として提示する。

実験の結果、各部位の抽出における“cvHaarDetectObjects”の精度は低かった。そのため、それぞれの部位について、座標情報を利用した制約条件を加えた実験を行った。設定した制約条件を Table 1 に示す。Table 1 における x, y は各部位の中心点の座標、width は顔の幅、height は顔の長さを示す。

Table1 顔の各部位抽出における制約条件

部位	条件
右目	$x < width/2 \ \&\& \ y > height/4 \ \&\& \ y < height/2$
左目	$x < width/2 \ \&\& \ y > height/4 \ \&\& \ y < height/2$
鼻	$x > width/3 \ \&\& \ x < width*2/3 \ \&\& \ y > height*2/5 \ \&\& \ y < height*2/3$
口	$x > width/3 \ \&\& \ x < width*2/3 \ \&\& \ y > height*3/4$

4.3 実装した顔認証機能の検証

4.3.1 検証内容

実装した顔認証機能を検証するために、本研究室の学生 20 名を対象に検証実験を行った。検証内容を以下に示す。

- 本研究室の学生 20 名に正面顔写真をそれぞれ 3 枚ずつ用意する.
- 60 枚の顔画像のパラメータを作成し、データベースを構築する.
- 60 枚のそれぞれの顔画像に対する顔認証候補を作成する.
- 顔認証候補の中に被写体の残りの 2 枚の正面顔写真が含まれているかを確認する.

4.3.2 検証結果

顔認証機能の結果を Table 2 に示す. なお, 顔認証候補の中に, 残りの 2 枚の写真が含まれている場合は, 順位の良いほうのみを結果に反映するものとする. 上記の

Table 2 顔認証機能の結果

候補順位	数
1	13
2	8
3	4
4	1
5	3
6	3
7	3
8	1
9	0
10	1

結果では, 60 枚中 37 枚について, 顔認証候補内に同様の被写体が確認されたことを示している. つまり, この確率は以下ようになる.

$$\frac{37}{60} = 0.6166.....$$

4.3.3 考察

任意の画像に対する顔認証候補として出る可能性にある画像は, 任意の画像を除いた枚数, すなわち 59 枚である. 59 枚からランダムに 10 枚を取り出し, その 10 枚の中に任意の画像と同様の被写体が写ったもの 1 つ以上がある確率は以下ようになる.

$$\frac{57C_8}{59C_{10}} + \frac{58C_9}{59C_{10}} \times 2 C_1 = 0.3652.....$$

これに対して, 今回得られた結果は, 0.6166..... であり, 今回作成した顔認証機能について, 一定の効果が出ていると考えられる. しかしながら, このような結果が出たのは, 顔認証の対象を正面顔写真としたからと考えられる. 実際の写真では, 顔は様々な方向を向いており, また様々な表情で写っている. このような写真では, 今回作成した顔認証アルゴリズムでの特徴点抽出の精度は低い. この精度は”cvHaarDetectObjects”関数そのものの精度の問題であり, OpenCV を利用した顔認証を行う限り改善することは困難であると考えられる.

今後, 本システムの改善策として, 画像以外の情報との連携が考えられる. 2.2.2 節で “登録画像のテキスト情

報書き込み機能”によって, 写真の中に写っている人の名前を入力することを説明した. この機能を応用することにより, A さんが写っている写真にはよく B 君が写っているといった統計的なデータの抽出が可能となる. この情報と今回実装を行った顔認証機能を連携させることにより, 機能向上が見込める.

5 まとめ

本報告では, 画像管理システム「ISDL Photo」の構築を行った. 本システムは, 画像の一覧表示, 画像の詳細表示, 画像への情報の付与, 画像内の顔認識, Exif 情報を利用した画像のカレンダー表示等の機能を持つ. また, 顔認識機能の拡張として, 顔認証機能の検討と実装を行った. 本システムを利用することにより, 研究室内の画像情報の有効活用, そしてそれらを介した研究室内での交流が行われることを期待する.

今後は, 研究室内での公開を視野に入れ, 現行のファイルサーバーへの保存形態を変更することのないシステムの構築を行うこと, そして顔認証機能の精度向上が課題となる. また, Exif 情報には撮影場所の GPS 情報などまだまだ有効活用出来る情報が含まれている. これら情報を基に他のサービスとの連携を目指す予定である.

参考文献

- 1) OpenCV Reference Manual
<http://www.cs.unc.edu/Research/stc/FAQs/OpenCV/OpenCVReferenceManual.pdf>
- 2) 矩形特徴による弱識別器のブースティングによる対象検出手法の汎化性能向上のための工夫と車載カメラの映像中の車の検出への応用
<http://www.kashin.risk.tsukuba.ac.jp/papers/2-1/ISS05-Tanigawa.pdf>
- 3) Haar Cascades
<http://alereimondo.no-ip.org/OpenCV/34>