New use for the pen: outline-based image queries *

Lambert Schomaker

Louis Vuurpijl
Nijmegen University/NICI, P.O. Box 9104,
6500 HE, Nijmegen, The Netherlands
Tel: +31-24-3616029, Fax: +31-24-3616066
schomaker@nici.kun.nl, http://hwr.nici.kun.nl/

Edward de Leau

Abstract

A method for image-based queries and search is proposed which is based on the generation of object outlines in images by using the pen, e.g., on color pen computers. The rationale of the approach is based on a survey on user needs, as well as on considerations from the point of view of pattern recognition and machine learning. By exploiting the actual presence of the human users with their perceptual-motor abilities and by storing textually annotated queries, an incrementally learning image retrieval system can be developed. As an initial test domain, sets of photographs of motor bicycles were used. Classification performances are given for outline and bitmap-derived feature sets, based on nearest-neighbour matching, with promising results. The next step is to use outlines for edge matching in raw, non-annotated images, for which preliminary results are presented. The benefit of the approach will be a user-based multimodal annotation of image database, yielding a gradual improvement in precision and recall over time.

Keywords: pen-based image queries, outline matching, edge matching

1 Introduction

In the search for image material in large databases, a number of query methods can be used, varying from keyword-based queries in textually annotated image databases to example-based and feature-based pictorial queries using the image content of the individual pictures (Table 1). On the world-wide web (WWW), various experimental approaches are already available[9], from which some lessons can be drawn.

Textual methods which are based on **keyword queries** (Table 1, A) to find images are potentially very powerful. However, a textual annotation of images produced by a single content provider, although already very costly by itself, usually does not cover a sufficient number of views or perspectives on the same pictorial material. Most importantly, however, the deixis problem is not solved: What is the exact location and area of a mentioned object in the actual image? This information may be relevant to the user, but is in any case extremely relevant to pattern classifiers in a machine-learning architecture.

In **text-based context search** (B), the query method also consists of keywords, but the annotation is automatically derived from the context of the image within the document[6, 8]. This approach may be brittle, because only in some domains (e.g., science, journalism) there exists something like a grammar for the juxtaposition of text and image which allows for finding meaningful correspondences between textual and pictorial content.

*Schomaker, L., Vuurpijl, L. & de Leau, E. (1999). *New use for the pen: outline-based image queries.* Proceedings of the 5th International Conference on Document Analysis and Recognition (ICDAR '99). Piscataway (NJ): IEEE. pp. 293-296.

Table 1. Types of queries and matching methods in image-based search

Query	*Matched with:*	*Matching algorithm*
A. textual (keywords)	manually provided textual image annotations	free text and information-retrieval (IR) methods
B. textual (keywords)	textual and contextual information in the image neighbourhood	free text and IR methods
C. exemplar image	image bitmap	template matching or feature-based schemes
D. layout structure (e.g., colored rectangles)	image bitmap	texture and color segmentation
E. object outline	image bitmap, contours	feature-based schemes
F. object sketch	image bitmap	feature-based schemes

The **example-based matching** methods (C) are usually disappointing for the user because the underlying goal of their query is not to look for "similar looking pictures", but for pictures "with similar object content". A fourth query method (D) consists of **layout specification**, for instance by means of placing variable-sized rectangles of different color and/or texture on a blank image, representing the query. Category (E), queries based on an **object outline** will be the focus of this paper. An outline is defined as a closed figure, drawn by the user around an object on a photograph by means of a pointing device (mouse or pen). A more difficult form of query is represented by (F), where the user is allowed to produce a free sketch of a figure which represents the visual query[4]. Combinations of the above query methods (A) to (F) can be used in a real application. The majority of the currently proposed methods are strongly characterized by a 'technology push'. However, it seems reasonable to derive some constraints from actual user demands before embarking on any development of pattern recognition algorithms. Basic questions are:

- are the users able to produce the queries?
- do the users like the query method?
- what level of classification performance will be acceptable to the user?
- is the system able to explain why a given match has been found?
- can the system learn from previous queries?

It is essential to exploit all known constraints, given the difficulties in content-based image retrieval. This means that user-context information such as user goals ("what is the user going to do with retrieved images?"), and system-related constraints (concerning, e.g., the user's software and hardware platform, bandwidth, etc.) should be used to improve the adequacy of the system response. In an on-line WWW survey on image-based search the following user responses were given (Table 2). From this table it can be inferred that users were less interested in color or texture, but reported to need images containing objects or other content. From this survey, it becomes apparent that users report to be mainly looking for an object in the image (122/170) and are much less interested in detailed image properties. Furthermore, photographic images seem to be more important than other types of graphic material: It was found that 68% of given responses concerned photographs[1]. These findings indicate that a successful image-based search method should be developed for photographic material, with a focus on object recognition methods. However, the seemingly effortless foreground/background segmentation in human visual perception is difficult to realize with current image processing and classification methods. Only under idealized conditions, bottom-up object segmentation based on edge detection and region analysis will be possible.

[1] Survey results were kindly provided by Arie Baris, a graduate student.

Table 2. User goals in image-based queries: objects, features or textures? Numbers represent the frequency of user responses in this on-line WWW survey (NA=no answer given). Respondents (N=170) were asked to respond to a number of questions related to recent image-based search actions they executed on the WWW.

Question	*Yes*	*No*	NA
"Did you need an image ..."			
"...with a particular object on it?"	122	41	7
"...with a particular color on it?"	25	137	8
"...with a particular texture on it?"	23	137	10

2 Method

The proposed method is based on a number of ideas, aimed at improving the classification performance and the usability of image-based search methods. The following points briefly describe the approach:

1. Focus on object-based representations and queries
2. Focus on photographic images with identifiable objects for which a verbal description can be given
3. Exploit the presence of human perceptual abilities in the user
4. Exploit human fine motor control by using a pen as the tool for drawing object outlines
5. Allow for incremental annotation of image material by storing user outline queries and annotations
6. Start with a limited content domain to evaluate these concepts

2.1 Focus on object-based representations and queries

Picard[5] makes a distinction between (a) subject, object and action search, (b) syntactic search (layout of images, breakpoints in video streams), (c) mood-related search, and the category (d) "I know what I'm looking for when I see it". Although looking for particular objects in images ("all horses", "mandolins") does not cover all possible forms of image needs in users, it is probably a very common user goal in an image-retrieval context, as evidenced from our survey, as well. Here, we define object as referring both to inanimate and animate objects in images.

2.2 Focus on photographic images with identifiable objects for which a verbal description can be given

Given the user preference for photographic images, it seems useful to focus the image-based retrieval efforts on this category. Since the purpose of the annotation concept presented in this paper is to bootstrap new image-retrieval methods which utilize both textual and pictorial query components, it is essential that a textual description of the object-based image query be added by the user.

2.3 Exploit the presence of human perceptual abilities in the user

In other areas of pattern recognition, the definition of particular classes may be relatively easy, apart from a manageable number of ambiguous cases, such as in speech or handwriting recognition. Moreover, in these fields,

well-known databases[1, 3] exist which contain truth labels of input patterns. The number of classes is typically limited to a few hundred unique patterns (i.e., characters, phonemes, visemes or words). In content-based image retrieval within an open domain, the number of classes is much larger, and there are not many public databases which are useful for the training of classification algorithms. Human assistance in multimedial annotation is essential in order to obtain a 'bootstrap collection' of object-based samples.

2.4 Exploit human fine motor control

The idea is to ask the user to produce queries by drawing an outline which encloses an object in a given image. The color of the outline should be contrastive with respect to the image content. Coordinates $(x(t), y(t))$ are recorded. The users are asked to follow the intended object boundaries with some precision. The outline should be a closed shape, and the user may have to guess its path at points of occlusion with other objects in the image. The computer mouse can be used but its accuracy and resolution are limited. A better solution is the electronic pen, in combination with 'electronic paper': an integrated digitizer and LCD screen. A problem may be the availability of a 'seed' image containing the object sought for. The solution is (a) to base the initial search on keywords, using a found image for further search, or, alternatively, (b) draw an outline by heart.

2.5 Allow for incremental annotation of image material

As in text-based information retrieval, almost every query can be considered as a valuable piece of condensed information which is based on genuine user goals and the user's understanding of the real world. This is especially true for multimedial queries in the form of object outlines. By asking the user to textually annotate the object outline of the query by using the keyboard, speech, or even handwriting recognition, a growing database of object outlines is formed, which is essential for the training of the image classification algorithms. There are other advantages of the proposed approach. The new standards MPEG-4 and MPEG-7 allow for an object-based image description. However, since object segmentation is difficult in an open image domain, there is a bottleneck at the point of creation of the object-annotated multimedia content. Pen-based techniques may be developed in order to alleviate this problem.

2.6 Start with a limited content domain to evaluate these concepts

Although the goals are high, we will constrain the image content domain first, to see whether the concept is fruitful. Images from a technological context have the advantage that a large number of object and object components can be identified, for which names do exist. The topic chosen here concerns a set of 200 mixed JPEG and GIF photographs of motor bicycles. Within this set, 750 outlines were drawn around image parts in the following classes: exhaust, wheels, engine, frame, pedal, fuel tank, saddle, driver, mirror, license plate, bodyworks, head light, fuel tank lid, light, rear light, totalling 15 object classes with 50 different outline samples of each object (Figure 3).

2.7 Algorithm

In our approach, a number of query scenarios can be envisaged. Here image-based queries can be done by matching in two representations: (a) Matching the query outline $(\vec{\mathbf{x}}, \vec{\mathbf{y}})$ with all outlines which are present in the database, and (b) matching the image $I(x, y)$ content within the outline $(\vec{\mathbf{x}}, \vec{\mathbf{y}})$ with existing templates in the database, (c) matching a query outline with image edges $\Delta I(x, y)$. Simple 1-NN matching will be used for all feature categories.

2.8 Features in outline-pattern matching

Based on handwriting recognition research - more specifically the recognition of isolated on-line handwritten characters [7] - we have developed the following feature set for outlines, to be used in image based queries.

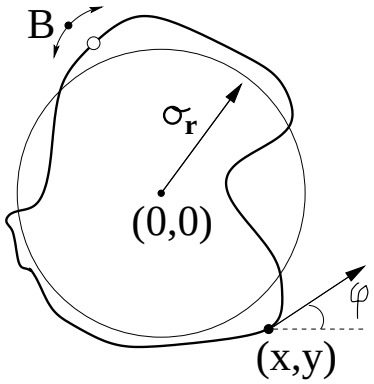

Figure 1. Features in outline pattern matching. The center of gravity is translated to $(0, 0)$, the size is normalized to an rms radius (σ_r) of one. From the starting point B, the matching process will try both clockwise and counter-clockwise directions, retaining the best result of both match variants. Other normalizations such as left/right or up/down mirroring are optional.

Figure 1 shows the used outline representation. A given closed raw outline (X_i, Y_i) with points $i = [1, N_r]$ which is derived from time-based measurements from a pointing device (preferably a pen) is resampled to a fixed and sufficient number of samples yielding (x_k, y_k) with points $k = [1, N_s]$. Given the limited bandwidth of the human motor system and the limited amount of time the user will invest in the production of an outline in a dynamical querying condition, the number of curvature maxima along the outline will be limited. Here we have chosen $N_s = 100$, which would coincide with about ten curvature peaks ($\mapsto$ ten ballistic strokes of 100ms) in real-time drawing behavior sampled at 100 points/second. This puts a soft upper limit to the complexity of contours. However, as an example, it is not necessary to produce, e.g., a meticulously shaped outline of a tree with all its leaves and tiny branches: a global approximation already contains useful information.

The center of gravity (μ_x, μ_y) is translated to $(0, 0)$, yielding (x'_k, y'_k). Then the standard deviation of all radii $r_k = \sqrt{((x'_k)^2 + (y'_k)^2)}$ is calculated yielding the rms radius σ_r. Finally, the outline is normalized to a radius σ_r of 1 by: $\hat{x}_k = x'_k/\sigma_r$ and $\hat{y}_k = y'_k/\sigma_r$ (again, assuming points $k = [1, N_s]$). The normalized outline $(\hat{x_k}, \hat{y_k})$ can then be used for scale and translation invariant matching. The resulting feature vector ($\vec{S_l}$) with all normalized x and y values can be compared with any other feature vector ($\vec{S_m}$) in a database by using the average squared Euclidean distance for simple nearest-neighbour matching:

$$\Delta_S = \frac{1}{N_s} \sum_{k=1}^{N_s} (S_{mk} - S_{lk})^2 \tag{1}$$

However, this feature vector will be not sufficient for accurate matching. In particular, what is missed are the curvature details along the curve. For this reason, a second feature vector $\vec{A}$ is defined, containing the running angle along the outline as $(cos(\phi), sin(\phi))$ (Figure 1). This feature vector contains more information about local changes in direction. Also for this feature vector, the distances between unknown and known outlines can be calculated, yielding Δ_A, similar to eq. 1. A third feature vector ($\vec{P}$) simply consists of the histogram of angles in the contour, i.e., the probability distribution $p(\phi)$, ϕ being bounded from $-\pi/2$ to $+\pi/2$. Also for $\vec{P}$, the

distances between a query and a template can be calculated, yielding Δ_P. The matching process entails two further provisions to implement invariance: (1) starting point (B, Fig. 1), (2) order and (3) horizontal mirroring normalization. Each outline query is matched with a sample outline, at a number of starting points along the curve, in a clock-wise and counter-clockwise fashion, both for a normal and a horizontally mirrored version. The best match, i.e., with the lowest distance, is kept in the hit list.

2.9 Within-outline image bitmap features

The following 68 features were derived from the pixels within the closed object outline:

color centroids The center of gravity for each of the RGB-channels. This gives 6 features: $R(x,y)$, $G(x,y)$ and $B(x,y)$

color histogram The histogram of the occurrence of 8 main colors: black, blue, green, cyan, red, magenta, yellow and white

intensity histogram A histogram for 10 levels of pixel intensity

RGB statistics The minimum and maximum values of each of the RGB-channels, and their average and standard-deviation (12 features)

texture descriptors A table of five textures was used, with five statistical features each (25 features)

invariant moments Seven statistical high-order moments[2] which are invariant to size and rotation

2.10 Matching outlines with detected edges

Ideally, pattern recognition methods will ultimately deliver object detection and classification in an open domain. Therefore, it is important to know the limits of bottom-up object detection. The human-generated outlines provide us with a useful reference set. Putting aside scale and translation problems, it is important to measure a match between a given object outline (of which we assume that the human perceiver correctly produced it) and edges which are computed using a generic approach. We have used the following distance measure. For each point i on a raw outline (X_i, Y_i), a convolution is calculated as follows. Let $\Delta I(x, y)$ be an estimate of the absolute and smoothed derivative of the luminance gradient of an image $I(x, y)$, averaged over a number of suitable directions. Then the local match between an outline point (X, Y) and the edge representation of the image can be calculated as:

$$M_{X_iY_i,\Delta I} = \sum_{\delta_x=-w}^{w} \sum_{\delta_y=-w}^{w} \frac{\Delta I(X_i + \delta_x, Y_i + \delta_y)}{\sqrt{\delta_x^2 + \delta_y^2}} \tag{2}$$

The kernel width parameter w will be chosen to limit the search to the direct neighbourhood of the outline pixel. The sum of M_{XY} of all points i along an outline $O = (X_i, Y_i)$ then is a measure of fit between the outline and the edge representation, which can be normalized by the number of points in the outline:

$$\mathcal{M}_{O,\Delta I} = \frac{1}{|O|} \sum_{i=1}^{|O|} (M_{X_iY_i,\Delta I}) \tag{3}$$

The measure $\mathcal{M}_{O,\Delta I}$ is a goodness of fit measure which, however, is sensitive to random speckles. By using an appropriate despeckling operator on the edge representations, only the sufficiently large connected components of edges will survive as potential candidates of the target object outline. In the set of n images $B = I_1, ..., I_n$, each image is annotated by k outlines $Q = O_1, ..., O_k$. In order to test the matching procedure, the outlines produced in a particular image are used as queries to find back the corresponding image in the image data base B. Two variants will be explored: (a) searching the best-matching image for a single outline, and (b), combining all k outlines in Q in a single image into a single query (Figure 2). This can be achieved by calculating $\mathcal{M}_{O,\Delta I}$ for each outline and averaging, yielding a combined measure of fit.

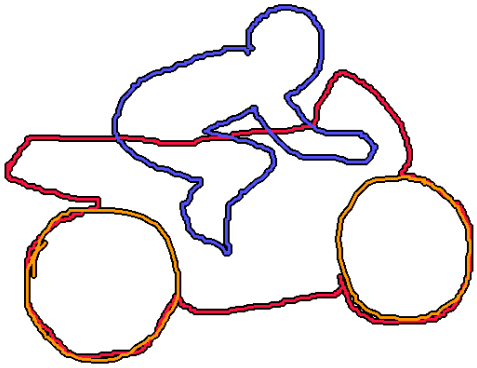

Figure 2. Several outlines drawn by a user onto a photograph of a motor cycle plus driver. There are three outlines: 1) driver, 2) frame, and 3) wheels.

3 Results

Table 3 (column I-III) shows the results for the normalized coordinates $(\hat{x}, \hat{y})$, the running angle $(cos\phi, sin\phi)$ and the histogram of directions $p(\phi)$. Results are expressed as average percentage of hits in a top-10 hit list. The outline coordinates $(\hat{x}, \hat{y})$ perform best. The four low-performance classes at the bottom of the table are the almost circular shapes without much difference in the outlines (the different lights and the fuel tank lid). The histogram of angles (column III) yields mediocre results. The rightmost column (IV) shows the results for the feature vector calculated from the image content within the outline curve. A selected subset of 29 out of 68 features was used based on stochastic optimization. It can be observed that even after such optimization, the within-outline image content is less reliable as a basis for object matching than the outline coordinates and running angle. Partly this is due to trivial factors such as color, partly these results may be caused by the fact that the image-based features are of a global nature (such as the invariant moments). More detailed analysis of precision vs recall (from P_1 to P_{50}) revealed no conflicting results.

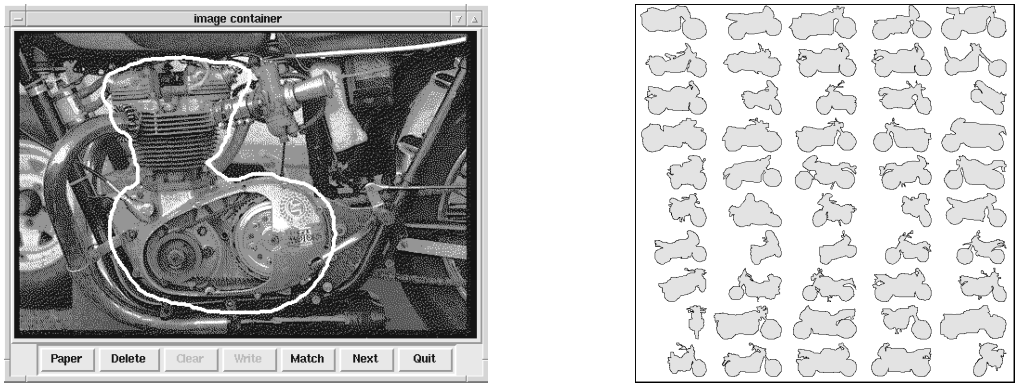

Figure 3. A query to find an *engine* (l) and a few outlines of "*frames*" (r)

Table 3. Classification performance of outline matching. Results are represented as the average percentage of correct hits in the top-10 hit list (P_{10}), averaged over $n = 50$ outline instances per class, of which each was used as a probe in nearest-neighbour matching. The query itself was excluded from the matching process. The total number of patterns is 750. Results for three groups of outline features and a group of imaged-based features are presented.

Query	I. P_{10} (%) $(\hat{x}, \hat{y})$	II. P_{10} (%) $(cos\phi, sin\phi)$	III. P_{10} (%) $p(\phi)$	IV. P_{10} (%) $image - based$
wheels	77.6	81.8	36.0	58.2
exhaust	75.4	79.4	34.0	34.6
engine	57.0	51.4	31.6	49.6
frame	52.0	33.8	38.8	69.4
pedal	47.4	47.2	22.8	33.0
driver	43.6	43.4	20.2	50.2
saddle	41.4	39.2	15.0	20.2
fuel tank	41.4	43.2	23.2	22.8
mirror	40.6	39.8	11.2	22.4
license plate	36.0	47.8	30.2	21.8
bodywork	31.0	26.6	14.4	22.4
head light	30.6	38.2	13.2	30.4
fuel tank lid	29.6	35.8	25.8	23.4
light	21.6	19.4	11.0	27.4
rear light	14.8	14.8	9.0	33.0

The following results were obtained for the match between the outlines and the bottom-up calculated image edge representations. It should be noted that a number of other matching criteria than M (eq. 3) were explored. The results presented here are achieved using 1) the best matching image to a query outline (*mean* in Figure 4) and 2) taking the best matching image to the complete set of outlines available for a query image (*max* in Figure 4). In the latter case, M was averaged over the number of sub components. On average, 3.3 outlines were present per image.

Using several outlines for an image query gives a better recall than using just one outline as can be observed in Figure 4. Note that whereas the top-0 performance is only 20%, this is an indication of the bottom-up performance. Although for the envisaged applications, a list of, e.g., 25 image thumbnails is acceptable, the performance is not sufficient to scale up to large image bases. Using more advanced object recognition techniques and techniques to filter out spurious edge intensities (e.g., edge-following or despeckling) improved results can be expected. Note, that in the current experiment, the kernel width parameter value w was set to one (yielding a 3x3 kernel) to prevent spurious hits on noisy edges.

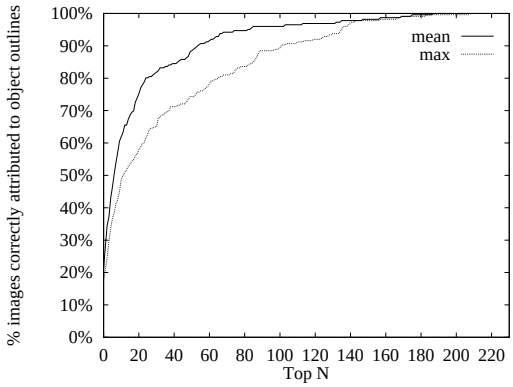

Figure 4. Results for the matching process between human-drawn outlines and bottom-up calculated image edges as a percentage of outline instances which are correctly associated with their original image. The two curves represent the results of sorting the hit list on mean convolution output $\mathcal{M}$ (solid line) or on the maximum value of $\mathcal{M}$ (stippled). This performance measure differs from Table 3 because here instances are matched as opposed to classes.

4 Discussion

Other matching procedures currently under study are: kNN and nearest centroid variants. A well-known property of image-based retrieval is the fact that the set of classes varies over time. For this reason, a neural-network solution for the problem as a whole is not suitable. However, for particular sub-domain problems, like face detection, specialized neural-network classifiers may be more attractive than simple distance-based schemes. Mostly, but not for all classes, the outline matching outperformed pixel-based matching. In any case will be useful to use the pixel content as a secondary constraint (e.g. on color or texture) in the search.

One of the reasons why this simple scheme without localized affine normalizations and/or perspective transform yields reasonable results may be the fact that photographers generate a limited number of 'canonical views' on objects, according to perceptual and artistic rules. Therefore, the number and range of camera attitudes towards the object, in this case a motor cycle, is limited.

Preliminary tests on matching human-drawn outlines with full-image edge representations showed promising results. Future work is directed at measuring the variation in outline-query shape and measuring the generalization. As we have seen, the generalization of an outline to other instances of the same outline class is reasonable. The goal is to ultimately obtain a similar performance for object outlines matched with the edge images of previously unseen instances.

Using the proposed system concept for the collection of a large number of object-based outlines, a training set will be created for the development of autonomous object classification. The availability of a large outline base for the training of an object-based image retrieval system at the level of both preprocessing (i.e., knowledge-based edge detection) and classification may ultimately result in considerable improvements in automatic object recognition in this application area. Combination of outlines in a particular geometric relationship will allow for more powerful queries than is possible on the basis of a single outline. Finally, the existing outline information can be used in the (Web) user interface to generate automatic highlighting of object components when the on-screen pointer moves over such components.

References

[1] Guyon, I., Schomaker, L., Plamondon, R., Liberman, R. and Janet, S.: Unipen project of on-line data exchange and recognizer benchmarks. Proceedings of the 12th International Conference on Pattern Recognition, ICPR'94, Jerusalem, Israel. IAPR-IEEE, (1994) 29–33

[2] Hu, M-K.: Visual Pattern Recognition by Moment Invariants. IRE Transactions on Information Theory **IT-8** (1962) 179–187

[3] Lamel, L.F., Kasel, R.H. and Seneff S.: Speech database development: Design and analysis of the acoustic-phonetic corpus. In Proceedings of the DARPA Speech Recognition Workshop (1987) 26–32

[4] Lopresti, D., Tomkins, A. and Zhou, J.: Algorithms for matching hand-drawn sketches. In: Downton, A.C. & Impedovo, S. (eds.): Progress in Handwriting Recognition. London: World Scientific (1997) 69–74

[5] Picard, R.W.: Light-years from Lena: Video and Image Libraries of the Future Proceedings of the International Conference on Image Processing (ICIP), Oct '95, Washington DC, USA. Vol I (1995) 310–313

[6] Rowe, N.C. and Frew, B.: Automatic caption localization for photographs on world wide web pages Information Processing & Management **34(1)** (1998) 95–107

[7] Schomaker, L.R.B.: Using stroke- or character-based self-organizing maps in the recognition of on-line, connected cursive script. Pattern Recognition **26(3)** (1993) 443–450

[8] Srihari, R.K.: Visually searching the Web for content. IEEE Computer **28(9)** (1995) 49–56

[9] WWW Reference page to image-based retrieval methods:
`http://hwr.nici.kun.nl/~profile/ibir/`