Raport Badawczy

Research Report

RB/32/2015

How to compare search engines in CBIR

T. Jaworska

Instytut Badań Systemowych
Polska Akademia Nauk

Systems Research Institute
Polish Academy of Sciences

POLSKA AKADEMIA NAUK

Instytut Badań Systemowych

ul. Newelska 6

01-447 Warszawa

tel.: (+48) (22) 3810100

fax: (+48) (22) 3810105

Kierownik Zakładu zgłaszający pracę:
Prof. dr hab. inż. Janusz Kacprzyk

Warszawa 2015

How to Compare Search Engines in CBIR?

Tatiana Jaworska
Systems Research Institute,
Polish Academy of Sciences
01-447, 6 Newelska Street, 01-447, Warsaw, Poland
Tatiana.Jaworska@ibspan.waw.pl

***Abstract*—At present a great deal of research is being done in different aspects of Content-Based Image Retrieval (CBIR) of which the search engine is one of the most important elements. In this paper we cover the state-of-the-art techniques in CBIR according to the aims of retrieval and matching techniques. The issue we address is the analysis of search engines reducing the 'semantic gap'. The matching methods are compared in terms of their usefulness for different user's aims. Finally, we compare our search engine with Google's and the SIFT method.**

***Keywords*— CBIR, search engine, SIFT, image query, image retrieval.**

I. Introduction

In recent years, the availability of large image datasets and search engines has increased tremendously. It is obvious that there is no universal CBIR system for finding all images and the spectrum of available systems ranges from the general purpose ones, like Google, to very narrowly specialized ones, like those found in medicine or astronomy. This multitude has necessitated a review in order to find the most suitable system for the user's purpose. The basic list of search engines is obtainable on the Internet [1].

Early on search engines used low-level features, such as colour, shape, texture information and annotations to retrieve similar images. This approach is still popular, but although many algorithms have been developed, they cannot adequately model image semantics and have many limitations when dealing with the vast resources of image databases. A survey on low-level image feature extraction in CBIR systems can be found in [2].

Hence, currently, the predominant engine categories are based on [3]:

- using object ontology to define high-level concepts,
- bag-of-visual-words (BoW), stemming from the text analysis,
- object retrieval using SIFT and its modification methods,
- introducing relevance feedback (RF) into a retrieval loop for continuous learning about users' intention,
- generating a semantic template (ST) to support high-level image retrieval,
- making use of both the visual content of images and the textual information obtained from the Web for WWW (the Web) image retrieval,
- combining visual properties of selected objects (or a set of relevant visual features), spatial or temporal relationships of graphical objects [4], [5], with semantic properties [6], [3].

The main contribution of this paper is the comparison of high-level semantic CBIRs with our new search engine which takes into account the kind and number of objects, their features, together with different spatial location of segmented objects in the image.

II. Aims of the Search Engine Construction

CBIR systems should meet the user's diverse requirements depending on the interest domain and the particular need. The user has to answer some questions of which the first and foremost is how to define their goal: do they want to construct a new CBIR system from scratch or build it on their existing image collections, for example, art collections, medical images, scientific databases or generally, the World Wide Web.

The next question which is inextricably connected with later selection criteria is whether there is a necessity of retrieval of whole images, some objects or maybe video fragments.

Another piece of required information is whether the annotations are assigned to the images in a DB. An answer to these problems will determine a single matching mechanism listed above as more efficient than the others.

Some other users need to make an order in their messy collection, while others want to find one object in many pictures, e.g. a face in an airport video, etc.

In the next subsection we will present advantages and disadvantages of the above-mentioned search engine categories..

III. Matching Techniques

A. Object Ontology

Generally speaking, ontologies define the concepts and relationships used to describe and represent an area of knowledge. Ontology gives the ability to model the semantics contained in images, such as objects or events. It provides, in a formal way, mutual understanding in a specific domain between humans and computers. Hence, ontology represents knowledge in a hierarchical structure which is used to describe and organize an image collection and it also shows the relation between these images.

In the early approaches high-level concepts were described using the intermediate-level descriptors of the object's ontology. These descriptors were automatically mapped from the low-level features calculated for each region in the database, thus allowing the association of high-level concepts and potentially relevant image regions [7]. Later, ontology was employed to spatial relationships in images such as connectivity, disjoint, meet, adjacency, overlap, cover, or inside. But the image was divided into 3x3, 5x5 or 9x9 windows instead of separate objects [8].

For ontological DBs the Web Ontology Languages (OWL), as a family of knowledge representation languages, have been constructed for authoring ontologies characterized by formal semantics.

An example of a search engine for multimedia has been proposed by Doulaverakis [9] and the system architecture is illustrated in Fig. 1. Here the user initiates a query by providing a QBE. This is depicted as case A in Figure 1 and comprises three steps. In the first step (1A) the content-based search is completed by analysing the provided multimedia content (i.e. performing the segmentation, extracting the low-level MPEG-7 descriptors and evaluating the distance between the prototype and the other figures stored in the multimedia database). The second step (2A) takes into account the metadata (which are mapped to the relevant ontologies) of the highest ranked results. For instance, the system may detect the highest ranked results in terms of visual similarity. Based on this information, an ontology-based query is formulated internally in the search engine, which links the knowledge base and enriches the result set with multimedia content that is close semantically to the initial content-based results (3A).

Eventually, the response returned to the user covers a wider range of items of interest, thus facilitating the browsing through the collection and shifting the burden of composing queries to the system instead of the user. The reverse process is equally interesting (case B in Fig. 1). Here, the initial query is a combination of terms defined in the ontology, e.g. 'Artefacts from the 1st century BC'. The knowledge base storing the ontology returns the items that fall into that category, as the first step (1B). The second step (2B) involves the extraction and clustering of the low-level multimedia features of this initial set, which is followed by multimedia retrieval, leading to the final step (3B).

Ontology is also a method for organizing extra large-scale image collections, like the ImageNet dataset, created at Stanford University [10].

There are some advantages of ontology:

- its application bridges the semantic gap;
- there is a special language for the user to ask a question;
- ontology-based algorithms are easy to design and are suitable for applications with simple semantic features.

The disadvantage is the necessity of preparing a special DB and annotating the introduction.

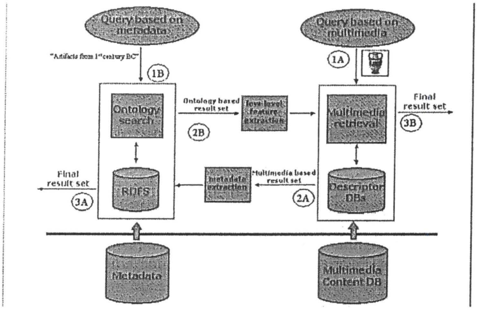


Fig. 1. A hybrid ontology and content-based search engine architecture follows [9].

B. Object Retrieval Using SIFT

The scale invariant feature transform (SIFT) was introduced by Lowe *[11]*, *[12]* to identify objects in two images, even if these objects were cluttered or under partial occlusion, because the SIFT feature descriptor is invariant to uniform scaling, orientation, and partially invariant to affine distortion and illumination changes.

An object is recognized in a new image by individually comparing each feature from the query to an image from a database and finding candidate matching features based on the Euclidean distance of their feature vectors. From the full set of matches, subsets of key points that agree on the object and its location, scale, and orientation in a query are identified to filter out good matches. Consistent clusters are determined by using an efficient hash table implementation of the generalized Hough transform. Each cluster of 3 or more features that agree on an object and its pose is then subject to further detailed model verification and subsequently outliers are discarded. Finally, the probability that a particular set of features indicates the presence of an object is computed through the Bayesian probability analysis, given the accuracy of the fit and number of probable false matches. Object matches that pass all these tests can be identified as correct with high confidence.

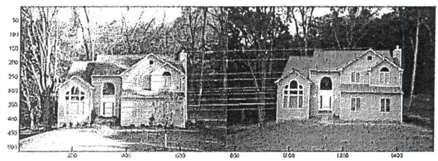


Fig. 2. Point-to-point correspondence found by the SIFT descriptors.

This property suggested that this method retrieves all images containing a specific object, even in a large scale image dataset, when that object is given as a query by example (QBE).

Hence, SIFT needs the query-by-example, but in some situations it may be difficult to provide, for instance, when we have an image in our mind but it is difficult to find it as QBE and additionally, we do not need the whole collection of similar images.

SIFT's additional advantage is the fact that it solved the problem of searching for disparity, independently of the issue of epipolar lines in stereovision. The example of point-to-point correspondence is presented in Fig. 2.

C. Bag of Visual Words

A simple approach to classifying images is to treat them as a collection of regions, describing only their appearance and ignoring their spatial structure which is very important in image representation. Similar models have been successfully used in the text community to analyse documents and are known as "bag-of-words" models, since each document is represented by a distribution over fixed vocabulary. Using such a representation, methods such as the probabilistic latent semantic analysis (pLSA) [13] and the latent Dirichlet allocation (LDA) [14] are able to extract coherent topics within document collections in an unsupervised manner.

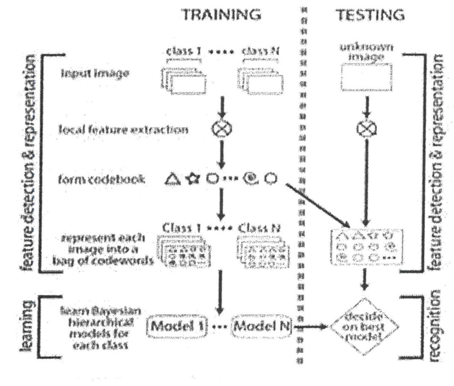


Fig. 3 Flow chart of the algorithm follows [15].

Some time ago, Fei-Fei and Perona [15] and Sivic et al. [16] applied such methods to the visual domain using [13] and [14] in their algorithm.

They model an image as a collection of local patches which are detected by a sliding grid and random sampling of scales. Each patch is represented by a code-word from a large vocabulary of code-words which are sorted in descending order according to the size of their membership and represent simple orientations and illumination patterns. By learning they achieved a model that best represents the distribution of these code-words in each category of scenes. In the recognition process they identified all the code-words in the unknown image. The training and testing process is presented in Fig. 3 in a symbolic way.

They found the category model that matched best the distribution of the code-words of the particular image. Their model is based on a principled probabilistic approach to learn automatically the distribution of code-words and the intermediate-level themes treated as texture descriptions.

An advantage of the BoW model that it is applicable in case of complex indoor and outdoor images. One of the notorious disadvantages of BoW is that it ignores the spatial relationships among the patches, which are very important in image representation. Additionally, the system needs the preparation of code-words, classes and Bayesian hierarchical models for each class.

D. Relevance Feedback

Large modern DBs actively employ user's interaction for relevance feedback (RF). This is an interactive technique based on feedback information between the user and a search engine in which the user labels semantically similar or dissimilar images with a query image, which is treated as positive and negative samples, respectively. Images labelled in this way are incorporated into a training set. The general architecture of such systems is presented in **Błąd! Nie można odnaleźć źródła odwołania.**.

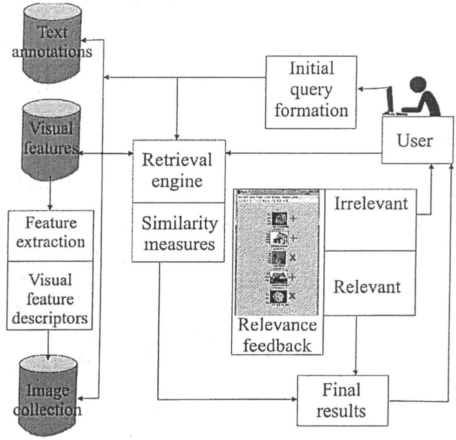


Fig. 4 CBIR architecture with the relevance feedback (RF) mechanism.

A more precisely labelled training set boosts algorithms to build a wider boundary between cluster features. For this purpose either Support Vector Machine (SVM) is applied to estimate the density of positive feedbacks or regarding the RF as a strict two-class on-line classification problem or discriminant analysis is used to find a low dimensional subspace of the feature space, so that positive feedbacks and negative feedbacks are well separated after projecting onto subspace.

During the last decade, various RF techniques have been proposed to involve the user in the loop to enhance the performance of CBIR [17], [18]. For example, L. Zhang et al training images are associated with only similar and dissimilar pairwise constraints, i.e., Conjunctive Patches Subspace Learning (CPSL) with side information, to explicitly exploit the user's historical feedback log data. It means that they minimize the distances between samples with similar pairwise constraints and to maximize the distances between samples with dissimilar pairwise constraints simultaneously. Samples are whole images for which neighbourhood is calculated as locally linear embedding (LLE) [20].

An option of RF is the adaptive technique based on the ostensive model of developing information needs proposed by J. Urban [21].

Generally, an advantage of RF approach is the fact that the system can start with a limited number of samples because the user will provide next labelled samples. RF has been proved to be effective in boosting image retrieval accuracy. The disadvantage is that most current systems requires about several iterations before it converges to a stable performance level, but users are usually impatient and may give up after two or three tries.

E. Semantic Template

In [22] Chang et al. introduced the idea of the semantic visual template (SVT) to link low-level image features to high-level concepts for video retrieval. A visual template is a set of icons or example scenes or objects denoting a personalized view of concepts such as meetings, sunsets, etc. The feature vectors of these example scenes or objects are extracted for the query process. To generate SVTs, the user first defines the template for a specific concept by specifying the objects and their spatial and temporal constraints, the weights assigned to each feature of each object. This initial query scenario is put to the system. Through the interaction with users, the system finally converges to a small set of exemplar queries that 'best' match (maximize the recall) the concept in the user's mind.

Firstly, the user submits a query image with a concept representing the image. After several iterations, the system returns some relevant images to the user. The feature centroids of these images are calculated and used as the representation of the query concept. Then the ST is defined as $ST = \{C,F,W\}$ with C the query concept, F the centroid feature obtained, and W being the weight applied to feature vectors. During the retrieval process, once the user submits a query concept, the system can find a corresponding ST, and use the corresponding F and W to find similar images.

A disadvantage of this system is the necessity of possessing a big lexical database [23].

F. WWW Image Retrieval

WWW search engines exploit the evidence from both the HTML text and visual features of images and develop two independent classifiers based on text and visual image features, respectively. The URL of an image file often has a clear hierarchical structure, including some information about the image, such as image category. In addition, the HTML document also contains some useful information in the image title, ALT-tag, the descriptive text surrounding the image, hyperlinks, etc.

However, the disadvantage is the fact that the retrieval precision is poor and as a result the user has to go through the entire list to find the desired images. This is a time-consuming process which always contains multiple topics which are mixed together. To improve the Web image retrieval performance, researchers are making an effort to fuse the evidence from textual information and visual image contents.

For example, Rasiwasia at al. proposed a combination of a query-by-visual-example (QBVE) with a query-by-semantic-example (QBSE) based on the probability of existance of a visual level represented as a set of feature vectors and the probability of a semantic concept by which an image is annotated. By using the Bayes rule and a similarity function based on methods measuring the distance between two probability distributions (such as the Kullback-Leibler Divergence, Jensen-Shannon Divergence, correlation, etc), they retrieve images most similar to the semantic signature [24].

On the other hand Wang et al. combine the visual features of images with the signatures received from the visual semantic space. For each relevant keyword, a semantic signature of the image is extracted by computing the visual similarities between the image and the reference classes of the keyword using the earlier trained classifiers. The reference classes form the basis of the semantic space of the keyword. If an image has N relevant keywords, then it has N semantic signatures to be computed and stored offline [25].

An advantage of the Web image retrieval is that some additional information on the Web is available to facilitate semantic-based image retrieval.

G. Our Search Engine with Combined Visual Properties

Our approach is more specific, more user oriented, and that is why we propose a special, dedicated user's GUI which enables the user to compose their ideal image from the image segments. The details of the system are described in [26] and [27].

The system concept is universal. In the construction stage we focus on estate images but for other compound images (containing more than several objects) other sets of classes are needed.

The main concept is presented in Fig. 5. In general, our system consists of five main blocks: the image preprocessing block [28], the classifying unit, the Oracle Database [29], the search engine [30] and the graphical user's interface (GUI). All modules, except the Oracle DBMS, are implemented in Matlab.

A classical approach to CBIR comprises image feature extraction [31], [32]. Similarly, in our system, at the beginning, the new image (e.g. downloaded from the Internet) is segmented, creating a collection of objects. Each object, selected according to the algorithm presented in detail in [28], is described by some low-level features f_i. We collect $r = 45$ features for each graphical object, for which we construct a feature vector $\mathbf{O} = \{f_1, f_2, \ldots, f_r\}$.

Next, the feature vector **O** is used for object classification. We have to classify objects in order to use them in a spatial object location algorithm and to offer the user a classified group of objects for the semantic selection. So far, four classifiers have been implemented and they are mutually used in our:

- a comparison of features of the classified object with a class pattern;
- decision trees [33], [34];
- the Naïve Bayes classifier [35], [36];
- a fuzzy rule-based classifier (FRBC) [37], [27].

The FRBC is used in order to identify the most ambiguous objects which means these assigned to different classes according the three first classifiers. According to Ishibuchi, the FRBC decides which of the three classes a new element belongs to [38].

Thanks to taking into account the spatial object location, the gap between low-level and high-level features in CBIR has diminished because by adding such crucial information, we can match images more efficiently and precisely.

To describe the spatial layout of objects, different methods have been introduced, for example: the spatial pyramid representation in a fixed grid [39], the spatial arrangements of regions [40], or the object's spatial orientation relationship [41]. In some approaches, image matching is proposed directly, based on spatial constraints between image regions [42].

Here, spatial object location in an image is used as the global feature [27]. The objects' mutual spatial relationship is calculated based on the centroid locations and angles between vectors connecting them, with an algorithm proposed by Chang and Wu [43] and later modified by Guru and Punitha [44], to determine the first principal component vectors (PCVs).

The data structure and the layout of the GUI reflect the manner of the search engine work. In order to help the user create the query which they have in mind, a special GUI has been prepared to formulate composed queries. First, the user selects a semantic concept by choosing of a line sketch and later they design their query. Some of such queries can be really unconventional as we can see in [28].

Now, we will describe how the similarity between two images is determined and used to answer a query. Let the query be an image I_q, such as $I_q = \{o_{q1}, o_{q2},\ldots, o_{qn}\}$, where o_{ij} are objects. An image in the database is denoted as I_b, $I_b = \{o_{b1}, o_{b2},\ldots, o_{bm}\}$. Let us assume that there are, in total, $M = 40$ classes of the objects recognized in the database, denoted as labels $L_1, L_2, \ldots, L_M$. Then, by the image signature I_i we mean the following vector:

$$\text{Signature}(I_i) = [\text{nobci}_1, \text{nobci}_2, \ldots, \text{nobci}_M] \qquad (1)$$

where: nobc_{ik} denotes the number of objects of class L_k present in the representation of an image I_i, i.e. such objects o_{ij}.

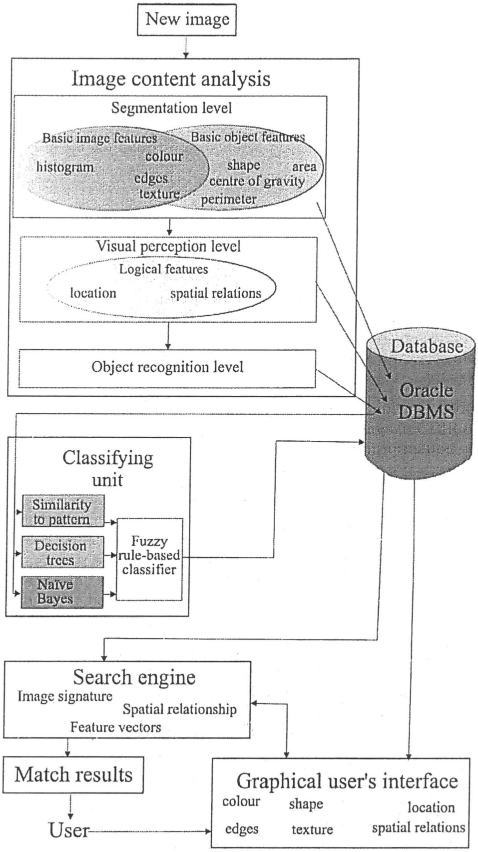


Fig. 5 Our content-based image retrieval system structure

In order to answer the query I_q, we compare it with each image I_b from the database in the following way. A query image is obtained from the GUI, where the user constructs their own image from selected DB objects. First of all, we determine a similarity measure sim_{sgn} between the signatures of query I_q and image I_b:

$$\text{sim}_{\text{sgn}}(I_q, I_b) = \sum_i (\text{nob}_{qi} - \text{nob}_{bi}) \qquad (2)$$

computing it as an analogy with the Hamming distance between two vectors of their signatures (cf. (1)), such that $\text{sim}_{\text{sgn}} \geq 0$ and $\max_i(\text{nob}_{qi} - \text{nob}_{bi}) \leq \text{tr}$, tr is the limit of the number of elements of a particular class by which I_q and I_b can differ. It means that we prefer images with the same classes as the query. Similarity (1) is non-symmetric because if some classes in the query are missing from the compared image the components of (1) can be negative.

If the maximum component of (1) is bigger than a given threshold (a parameter of the search engine), then image I_b is rejected, i.e., not considered further in the process of answering query I_q. Otherwise, we proceed to the next step and we find the spatial similarity $\mathrm{sim}_{\mathrm{PCV}}$ (2) of images I_q and I_b, based on the Euclidean, City block or Mahalanobis distance between their PCVs as:

$$\mathrm{sim}_{\mathrm{PCV}}(I_q, I_b) = 1 - \sqrt{\sum_{i=1}^{3}(PCV_{bi} - PCV_{qi})^2} \tag{3}$$

If the similarity (2) is smaller than the threshold (a parameter of the query), then image I_b is rejected. The order of steps 1 and 2 can be reversed because they are the global parameters and hence can be selected by the user.

Next, we proceed to the final step, namely, we compare the similarity of the objects representing both images I_q and I_b. For each object o_{qi} present in the representation of the query I_q, we find the most similar object o_{bj} of the same class, i.e. $L_{qi} = L_{bj}$. If there is no object o_{bj} of the class L_{qi}, then $\mathrm{sim}_{\mathrm{ob}}(o_{qi}, o_b) = 0$. Otherwise, similarity $\mathrm{sim}_{ob}(o_{qi}, o_b)$ between objects of the same class is computed as follows:

$$\mathrm{sim}_{\mathrm{ob}}(o_{qi}, o_{bj}) = 1 - \sqrt{\sum_{l}(Fo_{qil} - Fo_{bjl})^2} \tag{4}$$

where l is the index of feature vectors F_O used to represent an object. Thus, we obtain the vector of similarities between query I_q and image I_b.

$$\mathrm{sim}(I_q, I_b) = \begin{bmatrix} \mathrm{sim}_{\mathrm{ob}}(o_{q1}, o_{b1}) \\ \vdots \\ \mathrm{sim}_{\mathrm{ob}}(o_{qn}, o_{bn}) \end{bmatrix} \tag{5}$$

where n is the number of objects present in the representation of I_q. In order to compare images I_b with the query I_q, we compute the sum of $\mathrm{sim}_{ob}(o_{qi}, o_b)$ and then use the natural order of the numbers. Therefore, the image I_b is listed as the first in the answer to the query I_q, for which the sum of similarities is the highest.

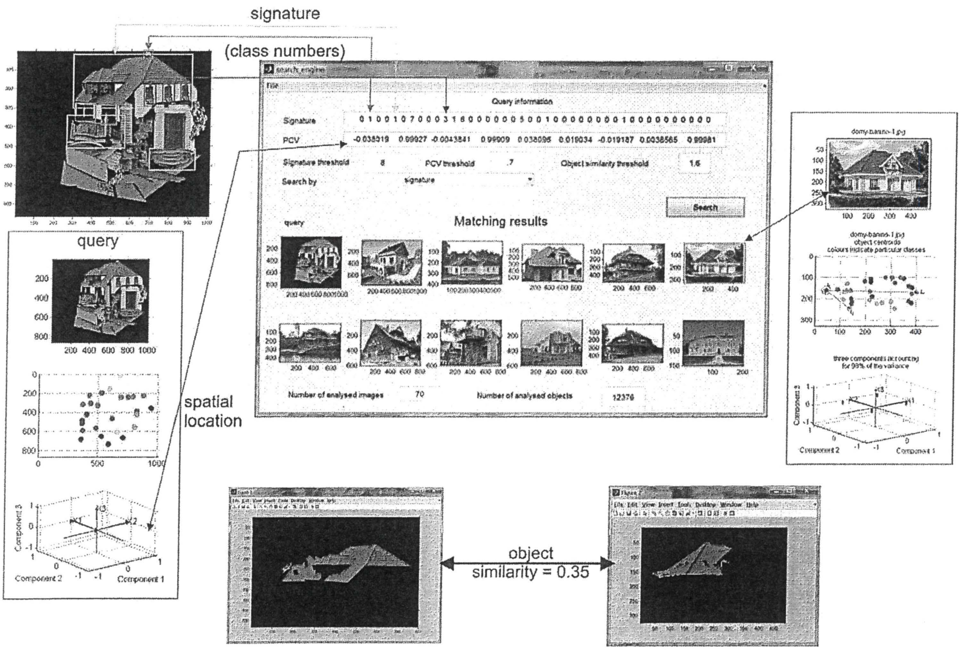


Fig. 6 A main concept of the search engine.

Fig. 6 presents the main elements of the search engine interface with reference images which are present in the CBIR system. The main (middle) window displays the query signature and PCV, and below it the user is able to set threshold values for the signature, PCV and object similarity. At this stage of system verification it is useful to have these thresholds and metrics at hand. In the final internet version these parameters will be invisible to the user, or limited to the best ranges. The lower half of the window is dedicated to matching results. In the top left of the figure we can see a user designed query comprising elements whose numbers are listed in the signature line. Below the query there is a box with a query miniature, a graph showing the centroids of query components and, further below, there is a graph with PCV components. In the bottom centre windows there are two elements of the same class (e.g. a roof) and we calculate their similarity. On the right side there is a box which is an example of PCA for an image from the DB. The user introduces thresholds to calculate each kind of similarity.

The strong side of our system is its semantic context which limits the semantic gap by taking into account middle-level features, such as objects, their numbers and spatial locations in an image. Additionally, we offer the user the GUI to compose their query by which we eliminate the necessity of looking for a QBE.

On the other hand, our system requires the preparation of DB containing objects, patterns, and classes

IV. Comparison Results

More than half of the new systems use a subset of the Corel image dataset [29] to test retrieval performance, others use either self-collected images or other image sets such as LA resource pictures [30]. The Corel image database contains 10,800 images from the Corel Photo Gallery divided into 80 concept groups, ranging from animals and outdoor sports to natural sceneries. These images are professionally pre-classified into different categories. Each group includes more than 100 homogeneous images. Some authors think that the Corel image dataset meets all the requirements to evaluate an image retrieval system, because of its large size, heterogeneous content and human annotated ground truth available.

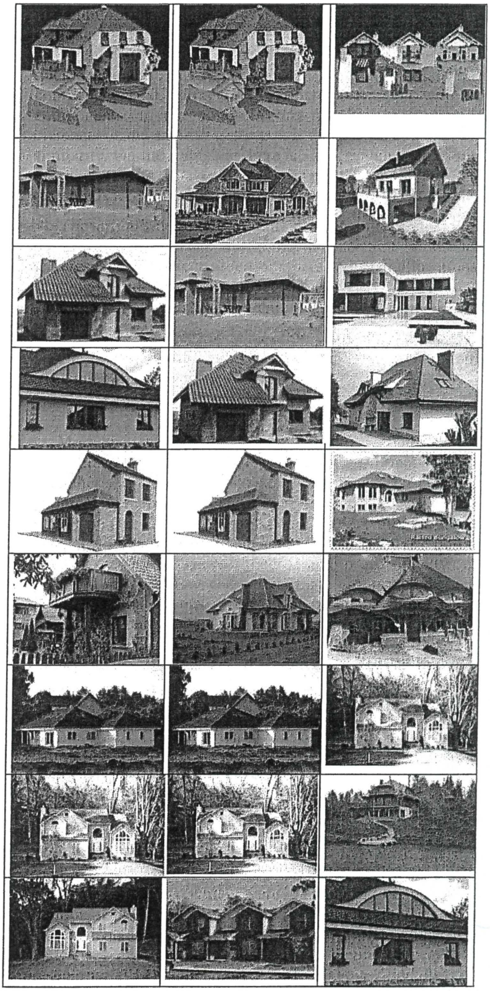
The Kodak database of consumer images [31], Brodatz textures [32], [33] are widely used in perceptual texture feature studies. Images collected from the Internet serve as another data source especially for systems targeting Web image retrieval [24] [25].

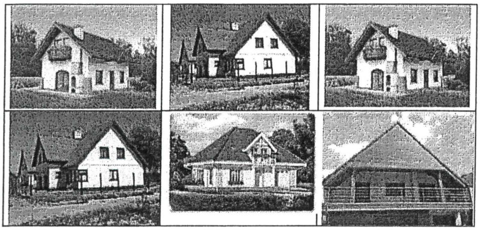
A. User Designed Query

We decided to prepare our own DB for two reasons: (i) when the research began (in 2005) there were few DBs containing buildings which were then at the centre of our attention and (ii) some existing benchmarking databases offered separate objects (like the Corel DB) which were insufficient for our complex search engine concept. At present, our DB contains more than 10 000 classified objects

As we have mentioned, a query is generated with the UDQ interface and its size, number of elements (patches) and complication depends on the user. The search engine displays a maximum of 11 best matched images from our DB. Although the user designed a few details, the search results are quite acceptable (see TABLE I).

TABLE I. The matching results for queries (in the first row) and the universal image similarity index for these matches when PCV similarity is calculated based on: (column 1) the Euclidean distance, (column 2) the City block distance (for thresholds: signature = 1





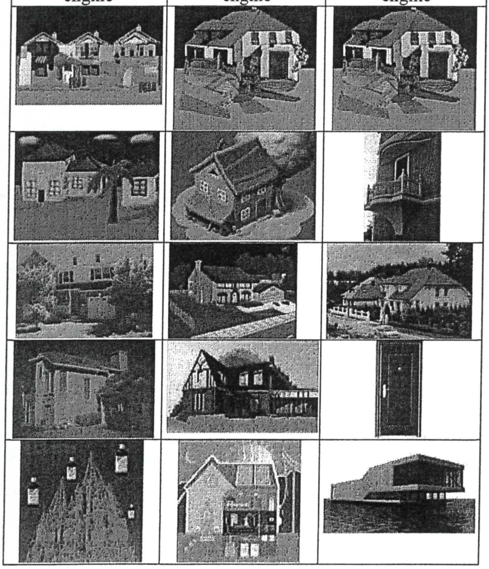
B. SIFT and the Google Image Search Engine

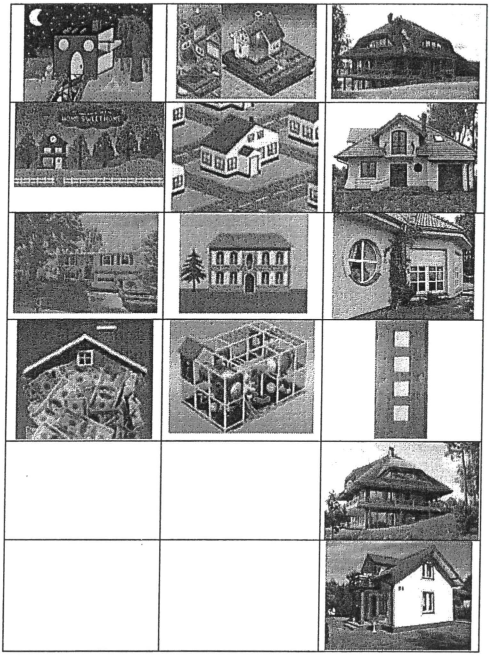
We also decided to compare our results with the Google image search engine. The results are presented in TABLE II. We also compare our search engine with the SIFT method and TABLE II column 3 presents the matching results for a query designed in our system. As it can be seen the best selected matches are those images whose elements can be found in the designed query.

We have opted for this comparison because these systems match images without annotations, which has been the most important condition. Systems using annotations belong to quite a different category while our focus is on pure image matching

TABLE II Matches for the Google and SIFT image search engine (Queries in the second row.)

The Google search engine	The Google search engine	The SIFT search engine





C. Discussion

The default comparison of search engines should be carried out based on the standard DB benchmarks. However, the user needs are more specific and we shall prepare dedicated search engines for these requirements, for instance, recognition of licence plate locations [51].

As we can see in Table II the Google engine treats the sketch houses as drawings, not as real photographs, whereas the SIFT one found the images from which the designed query consists, which is proper for this method, but has not been the user's intention who wants to receive house images most similar to their query in general and in detail.

V. CONCLUSIONS

The results presented here seem to be encouraging enough to move forward to the next stages of the CBIR system preparation, namely, to the GUI and the search engine. The methods already implemented will be also evaluated in terms of the addition of new classes to the system. GUI development will also enforce introducing subclasses to some of the most numerous classes.

Intensive computational experiments are under way in order to draw some conclusions regarding the choice of parameters for the search engine. However, the results we have obtained so far, using the simplest configuration, are quite hopeful.

As for the prospects for future work, the implementation of an optimised procedure should verify the feasibility of the approach. We expect a reasonable performance from the evaluation strategy outlined in the paper.

REFERENCES

[1] Wikipedia, "List of CBIR engines," 2015. [Online]. Available: http://en.wikipedia.org/wiki/List_of_CBIR_engines.

[2] F. Long, H. Zhang and D. D. Feng, "Fundamentals of content-based image retrieval," in *Multimedia Information Retrieval and Management Technological Fundamentals and Applications.*, New York, Sprainger-Verlag, 2003, pp. 1-26.

[3] Y. Liu, D. Zhang, G. Lu and W.-Y. Ma, "A survey of content-based image retrieval with high-level semantics," *Pattern Recognition*, vol. 40, pp. 262-282, 2007.

[4] S. K. Candan and W.-S. Li, "On Similarity Measures for Multimedia Database Applications," *Knowledge and Information Systems*, vol. 3, pp. 30-51, 2001.

[5] J. C. Cubero, N. Marín, J. M. Medina, E. Pons and A. M. Vila, "Fuzzy Object Management in an Object-Relational Framework," in *Proceedings of the 10th International Conference IPMU*, Perugia, Italy, 4-9 July, 2004.

[6] F. Berzal, J. C. Cubero, J. Kacprzyk, N. Marín, A. M. Vila and S. Zadrożny, "A General Framework for Computing with Words in Object-Oriented Programming.," in *International Journal of Uncertainty. Fuzziness and Knowledge-Based Systems.*, vol. 15 (Suppl), Singapore, World Scientific Publishing Company, 2007, pp. 111 131,.

[7] V. Mezaris, I. Kompatsiaris and M. G. Strintzis, "An ontology approach to object-based image retrieval," in *Proceedings of International Conference on Image Processing ICIP 2003.*, 2003.

[8] A. D. Gudewar and L. R. Ragha, "Ontology to Improve CBIR System," *International Journal of Computer Applications*, vol. 52, no. 21, pp. 23-30, 2012.

[9] C. Doulaverakis, E. Nidelkou, A. Gounaris and Y. Kompatsiaris, "A Hybrid Ontology and Content-Based Search Engine For Multimedia Retrieval," in *Workshop Proceedings in Advances in Databases and Information Systems ADBIS '2006*, Thessaloniki, 2006.

[10] O. Russakovsky and L. Fei-Fei, "Attribute Learning in Large-scale Datasets," in *Proceedings of the 12th European Conference of Computer Vision (ECCV), 1st International Workshop on Parts and Attributes.*, Crete, Greece, 2010.

[11] D. G. Lowe, "Object Recognition from local scale-invariant features," in *International Conferences on Computer Vision*, Corfu, Greece, 1999.

[12] D. G. Lowe, "Distinctive Image Features froam Scale-Invariant Keypoints," *Internationa Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.

[13] T. Hofmann, "Probabilistic latent semantic analysis," in *Proceedings of the15th Conference on Uncertainty in Artificial Intelligence*, Stockholm, 1999.

[14] D. M. Blei, A. Y. Ng and M. I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993-1022, 2003.

[15] L. Fei-Fei and P. Perona, "A Bayesian Heirarcical Model for Learning Natural Scene Categories," in *Computer Vision & Pattern Recognition CVPR*, 2005.

[16] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman and W. T. Freeman, "Discovering objects and their location in images," in *Proceedings of Internationa Conference of Computer Vision*, Beijing, 2005.

[17] L. Zhang, L. Wang and W. Lin, "Generalized biased discriminant analysis for content-based image retrieval," *IEEE Transactions on System, Man, Cybernetics, Part B - Cybernetics*, vol. 42, no. 1, pp. 282-290, 2012.

[18] L. Zhang, L. Wang and W. Lin, "Semi-supervised biased maximum margin analysis for interactive image retrieval," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2294-2308, 2012.

[19] L. Zhang, L. Wang and W. Lin, "Conjunctive patches subspace learning with side information for collaborative image retrieval," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3707-3720, 2012.

[20] S. T. Roweis and L. K. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science*, vol. 290, no. 5500, pp. 2323-2326, Dec. 2000.

[21] J. Urban, J. M. Jose and C. J. van Rijsbergen, "An adaptive technique for content-based image retrieval," *Multimedial Tools Applied*, no. 31, pp. 1-28, July 2006.

[22] S.-F. Chang, W. Chen and H. Sundaram, "Semantic Visual Templates: Linking Visual Features to Semantics," in *International Conference on Image Processing, 1998. ICIP 98.*, Chicago, 1998.

[23] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross and K. Miller, "Introduction to WordNet: An On-line Lexical Database," *Communications of the ACM*, vol. 38, no. 11, pp. 39-41 , Nov. 1995.

[24] N. Rasiwasia, P. J. Moreno and N. Vasconcelos, "Bridging the Gap: Query by Semantic Example," *IEEE TRANSACTIONS ON MULTIMEDIA*, vol. 9, no. 5, pp. 923-938, Aug 2007.

[25] X. Wang, S. Qiu, K. Liu i X. Tang, „Web Image Re-Ranking Using Query-Specific Semantic Signatures," *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, tom 36, nr 4, pp. 810-823, April 2014.

[26] T. Jaworska, "Multi-criteria object indexing and graphical user query as an aspect of content-based image retrieval system.," in *Information Systems Architecture and Technology*, L. Borzemski, A. Grzech, J. Świątek and Z. Wilimowska, Eds., Wrocław, Wrocław Technical University Publisher, 2009, pp. 103-112.

[27] T. Jaworska, "Spatial representation of object location for image matching in CBIR," in *New Research in Multimedia and Internet Systems*, vol. 314, A. Zgrzywa, K. Choroś and A. Siemiński, Eds., Wrocław, Springer, 2014, pp. 25-34.

[28] T. Jaworska, "Object extraction as a basic process for content-based image retrieval (CBIR) system," *Opto-Electronics Review*, vol. 15, no. 4, pp. 184-195, December 2007.

[29] T. Jaworska, "Database as a Crucial Element for CBIR Systems," in *Proceedings of the 2nd International Symposium on Test Automation and Instrumentation*, Beijing, China, 16-20 Nov., 2008.

[30] T. Jaworska, "A Search-Engine Concept Based on Multi-Feature Vectors and Spatial Relationship," in *Flexible Query Answering Systems*, vol. 7022, H. Christiansen, G. De Tré, A. Yazici, S. Zadrożny and H. L. Larsen, Eds., Ghent, Springer, 2011, pp. 137-148.

[31] T. Tuytelaars and K. Mikolajczyk, "Local Invariant Feature Detectors: A Survey," *Computer Graphics and Vision*, vol. 3, no. 3, p. 177–280, 2007.

[32] Y.-J. Zhang, Y. Gao and Y. Luo, "Object-Based Techniques for Image Retrieval," in *Multimedia Systems and Content-Based Image Retrieval*, S. Deb, Ed., Hershey, London, IDEA Group Publishing, 2004, pp. 156-181.

[33] U. M. Fayyad and K. B. Irani, "The attribute selection problem in decision tree generation," in *the 10th National Conference on Artificial Intelligence, AAAI*, 1992.

[34] J. R. Quinlan, "Induction of Decision Trees," *Machine Learning*, vol. 1, pp. 81-106, 1986.

[35] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack and D. Petkovic, "Efficient and Effective Querying by Image Content.," *Journal of Intelligent Information Systems*, vol. 3, pp. 231-262, 1994.

[36] I. Rish, "An empirical study of the naive Bayes classifier," in *IJCAI-2001 workshop on Empirical Methods in AI*, 2001.

[37] H. Ishibuchi and Y. Nojima, "Toward Quantitative Definition of Explanation Ability of Fuzzy Rule-Based Classifiers," in *IEEE International Conference on Fuzzy Systems*, Taipei, Taiwan, June 27-39, 2011.

[38] T. Jaworska, "Application of Fuzzy Rule-Based Classifier to CBIR in comparison with other classifiers," in *11th International Conference on Fuzzy Systems and Knowledge Discovery*, Xiamen, China, 19-21.08.2014.

[39] G. Sharma and F. Jurie, "Learning discriminative spatial representation for image classification," in *Proceedings of the British Machine Vision Conference 2011*, Dundee, 2011.

[40] J. R. Smith and S.-F. Chang, "Integrated spatial and feature image query," *Multimedia Systems*, no. 7, p. 129–140, 1999.

[41] X. M. Zhou, C. H. Ang and T. W. Ling, "Image retrieval based on object's orientation spatial relationship," *Pattern Recognition Letters,* vol. 22, pp. 469-477, 2001.

[42] T. Wang, Y. Rui and J.-G. Sun, "Constraint Based Region Matching for Image Retrieval," *International Journal of Computer Vision,* vol. 56, no. 1/2, pp. 37-45, 2004.

[43] C.-C. Chang and T.-C. Wu, "An exact match retrieval scheme based upon principal component analysis," *Pattern Recognition Letters,* vol. 16, pp. 465-470, 1995.

[44] D. S. Guru and P. Punitha, "An invariant scheme for exact match retrieval of symbolic images based upon principal component analysis," *Pattern Recognition Letters ,* vol. 25, p. 73–86, 2004.

[45] J. Deng, J. Krause, A. Berg and L. Fei-Fei, "Hedging Your Bets: Optimizing Accuracy-Specificity Trade-offs in Large Scale Visual Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Providence, RI, USA, 2012.

[46] Corel comp., "The COREL Database for Content based Image Retrieval".

[47] Z. Yang and C.-C. Jay Kuo, "Learning image similarities and categories from content analysys and relebance feedback," in *Proceedings of the ACM Multimedia Workshops. Multimedia00'*, Los Angeles, CA, USA , Oct 30 - Nov 03, 2000.

[48] the Eastman Kodak Company, [Online]. Available: http://r0k.us/graphics/kodak/.

[49] D.-C. He and A. Safia, "Multiband Texture Database," 2015. [Online]. Available: http://multibandtexture.recherche.usherbrooke.ca/.

[50] D.-C. He and A. Safia, "New Brodatz-based Image Databases for Grayscale Color and Multiband Texture Analysis," *ISRN Machine Vision,* vol. Article ID 876386, pp. 1-14, 2013.

[51] M. Koyuncu and B. Cetinkaya, "A Component-Based Object Detection Method Extended with a Fuzzy Inference Engine," in *Proceedings of the International Conference on Fuzzy Systems Fuzz-IEEE2015*, Istambul, 2015.