# Content-based Image Retrieval by Ontology-based Object Recognition

Jean-Pierre Schober, Thorsten Hermes, Otthein Herzog
{jps, hermes, herzog}@tzi.de

TZI Center for Computing Technology
Universitätsallee 21-23
28359 Bremen, Germany

**Abstract**

The main disadvantage of image retrieval systems is their lack of domain knowledge. Therefore a retrieval system has to focus on primitive features, as Eakins and Graham name them [3]. Due to the lack of background knowledge of the domain, the retrieval error rate is usually dissatisfying or the search options are limited to syntactic queries. Knowledge-based techniques allow for semantical searches filling the "semantical gap" [4]. In this paper we present a supervised learning system OntoPic, which is based on the well-known ontologies coded in DAML+OIL, for providing the domain knowledge. Combined with a DL reasoner for ontologies, the main target is to achieve a new level of result quality while allowing semantical searches. The main advantage of this approach is the usage of the reasoner as a classifier, enabling a dual use of the ontology. The same domain knowledge can be used for better object recognition, the basis for satisfying results, *and* a semantical search. Our work is applied to the domain of landscape images.

## 1 Introduction

It is even for a single person easy to obtain large amounts of images from the internet or through digital photography. To avoid a worthless archive of unorganized data a metadata annotation is necessary. However, good annotation is very costly, but if it is done it offers a high precision and recall and the value of the archive is multiplied.

There are many commercial and non–commercial applications which offer a solution to this problem. These so–called image retrieval systems are following mainly two different approaches [2]. The first approach offers searches for local or global image features, for example color or texture. The other approach follows the idea of adding keywords to the images as an annotation. Humans perform very well at annotating images, since they have normally a large knowledge about the domain the image belongs to. But besides the fact that it is a very annoying task to index a large amount of images, humans tend to subjectively annotate an image which invalidates this annotation effort for others [4]. Systems that follow the second approach offer support for the manual annotation or try to fully automate this process. The target is to minimize the subjectivity of manual annotation by guiding the annotation process or to make human assistance dispensable [2].

According to Eakings and Graham [3] image retrieval can be categorized into three levels: primitive, logical and abstract. The first retrieval approach presented in the previous paragraph moves only to the primitive, syntactic level, which is the lowest one. The second approach allows a search for logical objects of the image, and therefore fulfills a requirement for the higher, semantic levels. But as Eakins and Graham mentioned, this is not true *content–based* image retrieval, if humans provide the content information manually. Beside of this, the second approach is for most use cases[1] the superior one. Only the presence of annotated keywords allows for a so–called semantical search [2]. The main advantage of the semantical search is the fact that the user does not need to have a concrete idea of the image he is looking for. He only needs an idea of the context the image should belong to.

In this paper we present a supervised learning system, called OntoPic, which provides an automated keyword annotation for images and a content–based image retrieval on a semantical level.

## 2 Related Work

In this section we present some instances of image retrieval systems in chronological order. Three of the first approaches in content–based image analysis and retrieval in the early '90s were Photobook [11], the QBIC system (Query by Image Content) [10], and the IRIS (Image Retrieval for Information Systems) system [6]. QBIC offers a model search by means of single images and video sequences, considering only one data source in each case i.e., single images or videos. Photobook provides a set of interactive tools including interactive annotation capabilities for new images, for browsing and searching images and sequences. While the design of the QBIC system focuses on the idea that similarities between images should rather be defined exclusively by syntactical features,

[1]There are domains where a keyword indexing of images is not possible, due to their nature. Trademark logos are an example for this kind of images.

Photobook uses a category search based on text information associated to an image and a direct comparison of the images by the computation of so–called "Eigenimages" [11]. IRIS realized an automatic generation of content description of an image for special domains [6]. The project resulted in the ImageMiner system by IBM [8].

Breen et. al. promote an approach of integrating ontologies in the retrieval process [1]. In contrast to our approach the ontology is not fully integrated, but placed on top of the retrieval process. The main object classification is done by a neural network.

# 3 Reasoning about Ontologies

OntoPic has been developed at the Center for Computing Technologies at the University of Bremen. It is integrated into the PictureFinder[2] system which provides the basic feature extraction and segmentation capabilities for OntoPic. For the automated classification of image regions, ontologies are used to provide the needed domain knowledge. For this use case it is necessary that the ontology language provides reasoning support. OIL, respectively DAML+OIL, offers this reasoning support by a well defined mapping to the Description Logic (DL) $\mathcal{SHIQ}$ [7].

In the last section of this paper we discuss the capabilities of OntoPic. First we present the steps that are needed in OntoPic to extract content from a binary image. Further information about RACER and the semantics of the notions used throughout this paper is given in [5].

## 3.1 From Pixels to Meanings

OntoPic consists of three parts: a supervised training, an analysis, and a retrieval part. The first step to use OntoPic in an actual domain is the design of an ontology. Afterwards, this ontology can be trained with images from the concrete domain. When these steps are done, OntoPic is ready for use in that domain and can automatically analyze and annotate images. Analysed images can then be retrieved through queries by the users of the system.

The first step in designing an ontology is to define the hierarchy of the domain concepts. During the design of the ontology there are important design decision to be made, which can have a major impact on later classification results. The main point is to keep in mind, that the capabilities of the segmentation and feature extraction limit the usefull specialization grade.

[2]An online demo of the PictureFinder system is available under `http://www.tzi.de/bv/pfdemo/`

After a domain dependent ontology is drafted, it must be enriched, respectively trained, for the usage in the concrete domain. The trainer can load the ontology into OntoPic and train it by assigning ontology concepts to image objects.

## 3.2 Training Phase

After the trainer has assigned some images to the training set, he can either start an automatic color–region boundary detection for the images, mark the image boundaries directly by drawing into the image, or use a combination of these methods. During this step, it is vital to have the correct boundaries of the objects to avoid a false training of concepts. Therefore the trainer should always control and correct the results of the automated boundary detection to end up with a “semantic meaningful” segmentation [12]. After the training is completed, the ontology can be updated with the training results.

In order to do so, several local features are extracted and used for forming new concept axioms on *color*, *texture*, *background membership*, and *spatial relations*. The shape of a region is not yet taken into account, because of the difficult problems originating from the different viewpoints an image could have been taken.

## 3.3 Feature Extraction

The extracted features of a region are in general represented by continuous values. For a classification these values have to be discretized. The advantage is a smaller amount of concepts to be trained, as every trained concept instance represents a set of concepts covered by the discretized feature range. The disadvantage is the occurrence of overlapping in the feature space. OntoPic deals with this overlapping by applying background knowledge and is capable of discarding incoherent concept assignments, as described later.

### 3.3.1 Color, Texture and Background Membership

The color feature is based on the Color Naming System (cf. [9]). The color name is acquired by a conversion from RGB to HSB[3]. A color name is either a member of the set of chromatic or of achromatic color names. A chromatic color name is a composition of a saturation and lighting prefix with a description for the hue value, e.g., “very–light–vivid–green”.

A texture is either of kind *multiarea*, *homogeneous* or *speckled*. A texture of kind multiarea can be described additionally as *rippled* or *hatched*, and a

[3]The HSB color system is analogous the the HSV system, but provides a more intuitive description of colors.

speckled texture can be hatched, too.

The background membership is either *true* or *false*. As an indicator for this membership, the intersection of a region with the image border is taken.

Because of the features' different discriminating power they are weighted differently, too. The background membership for example, is the lowest weighted feature. The assignment of weights is done in a postprocessing step during the interpretation of the classification results.

### 3.3.2 Spatial Relations

Dealing with spatial relations in a two dimensional area is somewhat difficult and can easily yield to false categorization of image regions. In the landscape domain it is nearly impossible to find universally valid rules about spatial relations between different concepts. This is due to the nature of a photography which is in the worst case a reversed image. But these exceptions are negligible and have to stand back against the advantages that arise from a rule base. In the landscape domain the horizon is the line which divides sky elements from other landscape objects. Knowledge like "water is never above sky" can be very valuable and can help to avoid misclassifications.

There is often a correlation between two concepts concerning a spatial relation. As an example, an ocean is often right besides a beach and a lake besides grassland. These are no universal rules but valuable evidences, which can establish the difference between a right or wrong classification.

The following spatial relations are taken into account: *isAbove*, *isBelow* and *liesBeneath*.

## 3.4 Axiom–Building

After extracting the region features they can be used to build concept axioms for the manually trained concepts from the training set. The principle is to build a mapping between the low–level features and the high–level concepts via the DL.

### 3.4.1 Challenges and Solutions

The main problem of classifying objects via a DL is the need for an exact match with a formerly manually trained concept occurrence. For satisfying results, it is necessary to train nearly all different occurrences of one concept. This leads to the danger of strong overlappings in the feature space, an overspecification. Another problem arises from the fact, that the results of the feature extraction should be treated as uncertain until there are perfect feature extraction algorithms.

To avoid this problem, a degree of membership to a concept, a fuzzy logic approach is the classical solution. Unfortunately, there are no existing reasoning systems with the power of DL systems until now [13] which would cover this approach.

An approach to this problem is a pseudo–extension of the DL to a fuzzy logic or—to be more specific—a reduction of the fuzzy logic for use inside a DL [13]. The idea is to enrich the concept names with information about the degree of membership, resulting in a concept that we call *$\mu$–concept*. For example, the $\mu$–concept $\text{Tree}_{\geq 0.5}$ is interpreted as an instance of the concept Tree with degree $c \geq 0.5$. A parsing and interpretation of these concept names allows for an evaluation of the results. The logical relations between the different $\mu$–concepts have to be defined inside the ontology.

In our approach we do not use numbers to enrich the concepts, but identifiers for every feature. If a feature is the source for the belief that a region belongs to a concept, the identifier of the feature is added to the concept name. With this approach it is possible to classify image regions by a specific characteristic that was never trained before.

The three features color, texture and background are identified by the characters $C$, $T$ and $B$. A spatial relation is treated as a special feature as described in the next section. For every trained concept $CN$ it is necessary to auto–generate the following rules, which define the logical coherences between the enriched concept names[4]:

$$\begin{aligned} CN_{CT} &\doteq CN_C \sqcap CN_T \\ CN_{CB} &\doteq CN_C \sqcap CN_B \\ CN_{TB} &\doteq CN_T \sqcap CN_B \\ CN_{CBT} &\doteq CN_C \sqcap CN_B \sqcap CN_T \end{aligned}$$

### 3.4.2 Extending the Knowledge Base

After the training process is finished, the Terminological Box (TBox)[5] is extended by the training results. For every image object with corresponding feature identifiers $F_1 ... F_m$, feature roles $R_1 ... R_m$, feature values respectively role fillers $V_1 ... V_m$ and an assigned concept $CN$, the following statements are added to the TBox:

$$\begin{aligned} \exists R_1.V_1 &\sqsubseteq CN_{F_1} \\ &\vdots \\ \exists R_m.V_m &\sqsubseteq CN_{F_m} \end{aligned}$$

[4] We use the DL notation in this paper. More information about the semantics can be found in [5].

[5] The Terminological Box holds the general concept inclusions (GCIs). Together with the extensional knowledge in the Assertional Box (ABox) it forms a knowledge base.

The spatial relations of an object receive a special treatment. For every region the spatial relations to its neighbours are determined. In detail, a match for the spatial relation feature is given, if the region to be classified is in the same spatial relation to a neighbour as a formerly trained one.

## 3.5 Classifying an Image

For classifying a new image it has to be segmented into different image regions. Subsequently, for every region an ABox individual is created. The extracted region features are assigned via the proper role declarations to the individuals. To classify the regions, it is only necessary to let the reasoner realize the ABox and query for the individual direct types of the region instances. While parsing the enriched concept names it is possible to weight the results. For example, the concept $\text{Water}_{CT}$ is preferred over $\text{Sky}_T$ as the direct type of an individual, i.e. as the result of classification.

### 3.5.1 Non–Concepts/Postprocessing

In the previous section we described an approach to use a DL in combination with a reasoner for classifying image objects. Now we want to take advantage of the power that a reasoner gives us for the classification by using domain knowledge about spatial relations between the objects to end up with a consistent image classification.

At this point there exists a new problem: If we simply define axioms like "water is never above sky" we are likely to end up with inconsistent ABoxes. However, an inconsistency in the ABox is not the information we need, we need to know the cause of an inconsistency. Therefore, we introduce a new concept: the *non–concept*. Considering the prior example: "water above sky is non–water." We get a consistent ABox and we know that there is an inconsistency due to the presence of a non–concept *and* knowing the cause. An example for simple non–concept definition is:

$$\text{Water} \sqcap \exists \mathit{isAbove}.\text{Sky} \quad \sqsubseteq \quad \text{NonWater}$$

The existence of non–concept instances is interpreted as an inconsistency which is then solved by OntoPic externally. OntoPic removes the assignment from the individual, and it removes non–concept instantiations with the lowest degree of membership first. This process is repeated until the result is "consistent", i.e. contains no non–concept instantiations.

## 3.6 Retrieval

The retrieval process is also supported by the ontology. Due to the fact of the hierarchical organization of the ontology, it provides a thesaurus for user

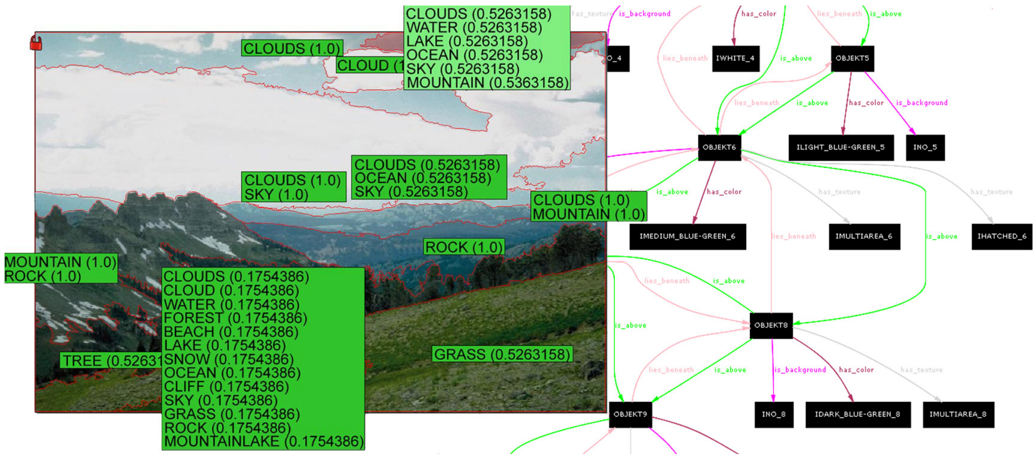


Figure 1: A result of the classification prior to the coherence check. The right image shows a part of the graphical representation of the underlying ABox.

queries. Furthermore, the ontology offers this hierarchy for the support of query formulations. Additionally, the domain dependent–knowledge can be combined to allow searching for scenes. For example, the knowledge base could hold the information that a sky, a beach, and water forms a beach scene.

# 4 Example

In this section we present an example to demonstrate the power of OntoPic. We use an ontology containing 71 concepts. The concepts of this ontology were trained with 75 images, leading to 450 assignments between concepts and image regions. Figure 1 shows on the left a classified image prior to the coherence check. On the right a graphical representation of the underlying ABox is shown.

There are three misclassified regions. The eight other regions are correctly classified. The misclassified region at the bottom has thirteen matches, all of them with a low degree of membership. Therefore OntoPic discards this result in the next step. The other two misclassified concepts have also multiple concept assignments. Some of these assignments are correct, others not. But as shown in figure 2, these wrong concept assignments are discarded during the coherence check due to applied domain–knowledge.

# 5 Conclusions

Ontologies are a powerful tool for describing domain knowledge. With the mapping to a DL ontologies are very useful for various applications.

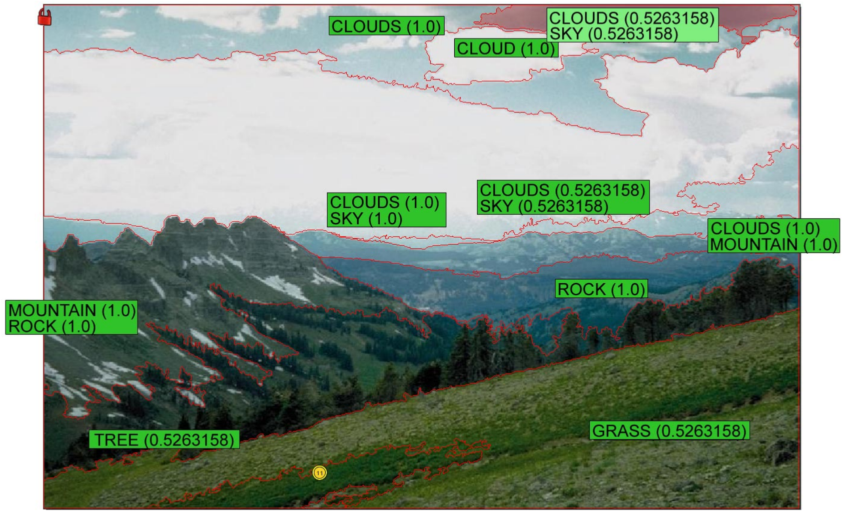


Figure 2: The final result, after the postprocessing/coherence check. The degree of membership is shown behind every concept name.

We have shown that it is possible to use a DL during the classification process and to benefit from the powerful reasoning capabilities RACER offers [5]. The DL offers the opportunity to use background knowledge about a specific domain and to raise the quality of the classification results.

Several problems were addressed and solutions proposed. It has been shown, that the capability to reason directly over a fuzzy logic would be of advantage for future applications.

Another feature of OntoPic that was not addressed in this paper is the aggregation of classified objects. OntoPic is capable of aggregating different parts of an object to by querying for the counterparts. This is possible due to the powerful query language which was implemented in RACER recently.

# References

[1] C. Breen, L. Kahn, A. Kumar, and L. Wang. Ontology-based image classification using neural networks. In *Proceedings of SPIE Internet Multimedia Management Systems*, III, pages 198–208, Boston, MA, USA, July 2002. SPIE.

[2] J. P. Eakins. Automatic image content retrieval – are we getting anywhere. In *Proceedings of Third International Conference on Electronic Library and Visual Information Research*, pages 123–135, May 1996.

[3] J. P. Eakins and M. E. Graham. Content-based image retrieval: A report to the JISC technology applications programme, January 1999.

[4] V. N. Gudivada and V. V. Raghavan. Content-based image retrieval systems. In *Proceedings of the 1995 ACM 23rd annual conference on Computer science*, number 28 in 9, pages 18–22, Nashville, Tennesse, USA, 1995. ACM Press.

[5] V. Haarslev and R. Möller. RACER user's guide and reference manual, 2004. Version 1.19. Hamburg, Germany: University of Hamburg, Computer Science Department, April 2004.

[6] T. Hermes, C. Klauck, J. Kreyß, and J. Zhang. Image retrieval for information systems. In W. Niblack and R. Jain, editors, *Storage and Retrieval for Image and Video Databases III*, volume 2420 of *SPIE Proceedings*, pages 394–405, San Jose, CA, USA, February 1995. SPIE.

[7] I. Horrocks. DAML+OIL: A description logic for the semantic web. *IEEE Data Engineering Bulletin*, 25(1):4–9, 2002.

[8] J. Kreyß, M. Röper, P. Alshuth, T. Hermes, and O. Herzog. Video retrieval by still image analysis with ImageMiner. In *SPIE Proceedings: Storage and Retrieval for Image and Video Databases V*, pages 236–247, San Jose, CA, USA, February 1997.

[9] J. M. Lammens. *A Computational Model of Color Perception and Color Naming*. PhD thesis, State University of New York, Buffalo, June 1994.

[10] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin. The QBIC Project: Querying Images By Content Using Color, Texture, and Shape. In *IS&T/SPIE Symposium on Electronical Imaging Science & Technology*, San Jose, CA, USA, February 1993.

[11] A. Pentland, R. W. Picard, and S. Sclaroff. Photobook: Tools for Content-Based Manipulation of Image Databases. In *SPIE Proceedings: Storage and Retrieval for Image and Video Databases II*, pages 34–47, San Jose, CA, USA, February 1994.

[12] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, December 2000.

[13] U. Straccia. Reducing fuzzy description logics into classical description logics. Technical Report 2004-TR-06, ISTI-CNR, Pisa, Italy, February 2004.