PICTUREFINDER: DESCRIPTION LOGICS FOR SEMANTIC IMAGE RETRIEVAL

Jean-Pierre Schober, Thorsten Hermes, Otthein Herzog

Center for Computing Technologies
University of Bremen, Germany

ABSTRACT

Large amount of images need an efficient way of retrieving them. The usual approach of manually annotating images and/or providing a syntactic retrieval capability lacks flexibility and comfort. The automatic annotation of images is a main target of the image retrieval community. These so called content-based image retrieval (CBIR) systems focus on primitive features, as Eakins and Graham [1] name them. Description logics (DL) offer a useful contribution to content-based image retrieval while allowing logical reasoning about the semantic contents of the image and ending with consistent classification results. This is a main advantage about traditional classification algorithms. Another advantage is the possibility to use domain knowledge, which is formulated in DL, on the retrieval side, thus offering a semantic retrieval. In this paper we present an approach and the results of adopting a DL for classifying image regions.

1. INTRODUCTION

Today, it is even for a single person easy to obtain large amounts of images from the internet or through digital photography. To avoid worthless archives of unorganized data, metadata annotation is necessary. However, good annotations are very costly, but when done they offer a high precision and recall and the value of certain archives is multiplied.

According to Eakings and Graham [1] image retrieval can be categorized into three levels: primitive, logical and abstract. Many retrieval systems move only to the primitive, syntactic level, which is the lowest one. More advanced systems allow a search for logical objects of the image, and therefore fulfill requirements for the higher, semantic levels. The semantical search is in most cases[1] the superior one.

In order to provide this it is more than useful to understand the image content. To understand the true content of an image it is necessary to have knowledge about the domain the image belongs to and to be capable to reason about this knowledge. Description logics (DL) provide an efficient way of managing knowledge and to do reasoning steps over it. The integration of a DL into object recognition can be done roughly in two ways: fully integrated or with an on–the–top–approach. The latter one requires a traditional classification of image regions with the DL placed on top, doing verification of the results. Our approach consists of the integration of the DL into the classification process, namely doing the classification with a DL. The target is to end up with consistent classification results due to the adaption of domain knowledge, e.g. constraints about spatial relations.

In this paper we present a supervised learning system, called OntoPic, which provides an automated annotation for images in the domain of landscapes, and therefore provides a content–based image retrieval on a semantical level. It is obvious, that our approach allows an easy adaption to other domains.

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

2. THE CBIR-SYSTEM PICTUREFINDER

PictureFinder has been developed at the Center for Computing Technologies at the University of Bremen (cf. [2]). The module that is responsible for the high–level annotation of images is called OntoPic.

The rough principle of OntoPic is sketched as follows: An expert models a domain depended ontology and trains it, by assigning prototypical image regions to concepts of the domain. The system automatically extracts the features of the assigned regions and extends the knowledge base with rules which map the feature values to the concept. An ontology enriched in this way can afterwards be used for an automatic annotation of an image by classifying the regions of this image. Therefore the system can be divided into two parts: Training and analysis.

Instead of using a DL for modelling the domain knowledge, we use more common ontology languages. This is possible, due to the fact that languages like DAML+OIL provide a well–defined mapping to the DL $\mathcal{SHIQ}$ [3]. As a reasoner for DL we use RACER. Further information about RACER and the semantics of the notions used throughout this paper is given in [4].

2.1. Training the ontology

The target of the training part is to build a mapping between the syntactic and semantic level. The image regions can be extracted automatically by a color segmentation, and treating areas with similar colors as image regions. To ensure a "semantic meaningful" segmentation [5] the trainer can manually correct the results of the automated segmentation during this training step. The vital part of the training is the assignment of segmented image regions to concepts of the domain.

2.2. Feature Extraction

To handle the continuous feature values inside the DL we discretize these values. We deal with the known disadvantages of the disrectization, like overlappings in the feature space, by applying knowledge about spatial constraints. Using these constraints it is possible to discard incoherent concept assignments, as described later.

2.2.1. Color, Texture and Background Membership

For discretizing color the RGB value is converted into HSB[2] color space and then transformed into the Color Naming System (CNS, cf. [6]). The CNS describes colors with natural language names as 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 as *rippled* or *hatched*, and a speckled texture additionally as hatched.

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 also weighted differently. 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.

2.2.2. Spatial Relations

When dealing with spatial relations one has to distinguish between two different kinds. The first are universally valid, the second are only common spatial relations that occure in images of the specific domains. The first one can define spatial constraints, while the latter can help by classifying image regions. The contribution to the classification is based on the fact, that there is often a correlation between two concepts concerning a spatial relation. As an example, an ocean can be found beside a beach and a lake beside grassland. These are not 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*.

[2]The HSB color system is analogous to the HSV system.

2.3. 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.

2.3.1. Challenges and Solutions

To avoid the problem that every concept must be trained in every prototypical occurence a fuzzy logic approach is the classical solution. Unfortunately, there are no existing reasoning systems with the power of DL systems until now [7] which would cover this approach.

To cope with this problem our approach is to a pseudo–extension of the DL to fuzzy logic or—to be more specific—a reduction of the fuzzy logic for use inside a DL [7]. The idea is to enrich the concept names with information about the degree of membership, resulting in a concept that we call *μ–concept*. For example, the μ–concept 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 μ–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.

The 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 statements, which define the logical coherences between the enriched concept names:

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

The first definition states that every instance of the concept CN_{CT} is an instance of the concepts CN_C and CN_T and vice versa. Detailed information about the semantics can be found in [4].

2.3.2. *Extending the Knowledge Base*

For each concept the trainer has assigned to an image region, the Terminological Box (TBox)[3] is extended by mapping from the image region features to the concept. Formally, for every image region 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:

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

With these statements it is ensured that a region with the value V_1 for the feature role R_1 is an instance of the concept CN_{F_1}. The following statements are an example for the assignment of a water region to the concept water:

$$\begin{aligned} \exists\, hasColor.\text{blue} &\sqsubseteq \text{Water}_C \\ \exists\, hasTexture.\text{homogeneous} &\sqsubseteq \text{Water}_T \\ \exists\, isBackground.\text{true} &\sqsubseteq \text{Water}_B \end{aligned}$$

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.

2.4. Classifying an Image

The classification of segmented image regions is the main task of the DL. Subsequently, for every region an individual is created inside the ABox. The extracted region features are assigned via the proper role declarations to these individuals. To classify the regions, it is only necessary to let the reasoner query for the individual direct types of the region instances. The direct types of an individual are the most specific concepts an individual is instance of. The result of this classification are enriched concept names (like Water$_{CT}$). The indices of these enriched concept names can be interpreted as the feature matches, responsible for the classification result. This allows a weighting of the results. For example, the concept Water$_{CT}$ is preferred over Sky$_{TB}$ as the direct type of an individual, because the color match has a bigger weight than the texture match.

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

2.5. Non–Concepts/Postprocessing

The target of this postprocessing step is to end with a consistent classification of the image regions. Consistency has to be understood with respect to spatial constraints. Instead of simply defining rules like "water is never above sky", which leads to an inconsistency in the ABox, we introduce a new concept type which we name *non–concept*. The occurence of such non–concepts in a classified image has to be understood as a contradiction which has to dissolved externally. The following example should clarify the use of non–concepts: The non–concept *non–water* is defined as a water concept which lies above a sky concept:

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

If we classify an image region as water and this region lies above a sky region the reasoner can deduce that there is an inconstistency in the classification result. It has to be mentioned, that in the former example the sky region as well would be classified as *non–sky*. By removing non-concept instantiations, starting with the lowest degree of membership, such inconsistencies can be resolved. An image classification is therefore treated as consistent if there are no occurences of non–concepts. Some example images are shown in [8].

2.6. Retrieval

The retrieval process is also supported by the ontology. Due to the hierarchical organization of the ontology, it provides a thesaurus for user queries. Furthermore, the ontology offers this hierarchy for the support of query formulations. Additionally, the domain dependent–knowledge can be combined to allow for the search of scenes. For example, the knowledge base could hold the information that a sky, a beach, and water forms a beach scene.

3. RESULTS

For measuring the result quality a set of 85 different images from the landscape domain was taken. 30 images were separated and used as training images. Multiple independent training steps were done, each which a different amount of used concepts and assigned regions per concept. The target of these training steps was to verify, how many concepts are distinguishable by our approach.

The graphs in the figures above show the precision and recall values for a set of five, ten and fifteen differently trained concepts. As expected, the ideal amount of concepts is low—between 5 and 10. The reason for this is based on the fact, that there exists different concept classes which we call base and logical concepts. Base concepts are mainly defined through their syntactic attributes like a specific color

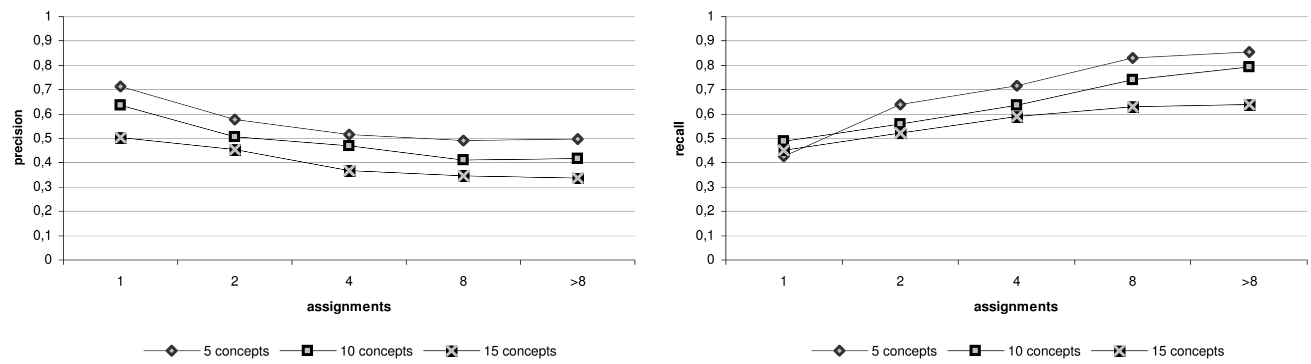

Fig. 1. Precision and recall for different numbers of assignemnts per concept and sets with a different concept amount.

or texture. Whereas a logical concept is defined through its relation to other base concepts and the context of the image. For example the concept sand is a base concept but the concept beach is a logical concept, which—as a specialization of sand—can only be distinguished from the concept sand by the context, e.g. the spatial relation to a nearby ocean. As a consequence of this, it is necessary to map only the base concepts to image regions.

It has to be mentioned, that it is possible to derive logical concepts from a set of base concepts. Therefore an annotation only consisting of base concepts is a kind of "semantic preserving image compression" [9]—regarding the annotation.

It has be shown, that it is sufficient to assign two prototypical regions per concept to get the best tradeoff between precision and recall. More assignments per concept lead to a better recall while minimizing the precision. One reason for this result is the fact, that a higher number of assignments raises the probability of assigning non–prototypical regions.

4. FURTHER WORK

As a consequence of this project the next step is to define a mechanism which automatically falls back to base concepts if a logic concept is trained directly. The system should only apply logical rules—like constraints about spatial relations—to logical concepts. This improvement should hopefully lead to a state, where the system is capable of learning from spatial relations of logical concepts.

5. REFERENCES

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

[2] T. Hermes, A. Miene, and O. Herzog, "Graphical search for images by PictureFinder," *Int. J. Multimedia Tools and Applications. Special Issue on Multimedia Retrieval Algorithmics*, 2004.

[3] I. Horrocks, "DAML+OIL: A description logic for the semantic web," *IEEE Data Engineering Bulletin*, vol. 25, no. 1, pp. 4–9, 2002.

[4] 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.

[5] 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*, vol. 22, no. 12, pp. 1349–1380, Dec. 2000.

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

[7] U. Straccia, "Reducing fuzzy description logics into classical description logics," Tech. Rep. 2004-TR-06, ISTI-CNR, Pisa, Italy, Feb. 2004.

[8] J.-P. Schober, T. Hermes, and O. Herzog, "Content-based image retrieval by ontology-based object recognition," in *Proceedings of the KI-04 Workshop on Applications of Description Logics (ADL-2004)*, V. Haarslev, C. Lutz, R. Möller, and S. Bechhofer, Eds., Ulm, Germany, Sept. 2004.

[9] 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*, San Jose, CA, USA, Feb. 1994, pp. 34–47.