A GENERIC CONCEPT FOR OBJECT-BASED IMAGE ANALYSIS

André Homeyer, Michael Schwier and Horst K. Hahn
Fraunhofer MEVIS, Institute for Medical Image Computing, Universitätsallee 29, Bremen, Germany

Keywords: Image analysis, Image understanding, Classification, Attributed relational graph, Database.

Abstract: Object-based image analysis enables the recognition of complex image structures that are intractable to conventional pixel-based methods. To date, there is no generally accepted approach for the object-based processing of images, thus making it difficult to transfer developments. In this paper, we propose a generic concept for object-based image analysis that is broadly applicable and founded on established methodologies, such as the attributed relational graph, the relational data model and statistical classifiers. We also describe a reference implementation of the concept as part of the MeVisLab image processing platform.

1 INTRODUCTION

Many problems in computer vision require the analysis of image structures on the basis of their properties and mutual relations. This requires strategies for the extraction and abstract representation of the image content. The common pixel-based representation of images is usually insufficient because the artificial discretization of the image does not reproduce its semantic entities. Therefore, we propose a region-based analysis concept.

By partitioning an image into regions, a structuring is gained which corresponds to the way humans comprehend an image. Regions exhibit a wealth of meaningful features (like texture, shape and spatial context) that single pixels lack. In this manner, the region-based representation of images simplifies the application of prior knowledge to their analysis.

The scientific discipline of the semantic processing of image regions is commonly called *object-based image analysis*. An object, in this regard, is an abstract representation of a single image region and its properties. In the past years, object-based image analysis has become a common practice in the field of geographic information science. For instance, in (Shackelford and Davis, 2003), object-based image analysis is utilized for urban land cover classification from high-resolution multispectral image data. However, it has shown promise in other disciplines as well. Hay and Castilla (Hay and Castilla, 2006) list the strengths and opportunities of object-based image analysis as well as its possible weaknesses and threats.

To date, there is no generally accepted model for the object-based representation of images. Therefore, we propose a generic concept for processing and managing images on the basis of objects, that is supposed to be broadly applicable and complementary to existing pixel-based image processing methods. In contrast to proprietary solutions like (Schäpe et al., 2003), the proposed methodology is entirely founded on established and commonly known concepts, such as the attributed relational graph, the relational data model and statistical classifiers.

In the following sections, we outline our concept for object-based image analysis and give a short overview of a reference implementation on the basis of the MeVisLab image processing platform.

2 CONCEPT

For representing knowledge that is extracted from the image or acquired through reasoning, the proposed concept incorporates a well-known data model—the *attributed relational graph*. Attributed relational graphs are abstract networks in which both nodes and edges are labelled with numerical or nominal attributes. Although attributed relational graphs are simple and intuitive data models, they are capable of representing complex structures and have been successfully used for modelling image content before (Aksoy, 2006; Chang and Fu, 1979).

Homeyer A., Schwier M. and K. Hahn H. (2010).
A GENERIC CONCEPT FOR OBJECT-BASED IMAGE ANALYSIS.
In *Proceedings of the International Conference on Computer Vision Theory and Applications*, pages 530-533
DOI: 10.5220/0002848105300533

Every node in the graph corresponds to a single *image object*, that is, one or multiple connected regions in the image. Image objects can arbitrarily span across the spatial dimensions of the image. For each image object, there has to exist a unique mapping to the corresponding pixels in the image. In the most basic case, this mapping is established through a labeled image in which pixels from one object store a common integer value identifying the object. In addition to this kind of *direct* mapping, objects can be mapped *indirectly* through directly mapping objects which constitute their parts.

Edges in the attributed relational graph represent contextual *relations* between image objects, such as adjacency, overlap and parthood. While the meaning of relations is typically subject to the application, there is a set of generally defined relations that are commonly used. Examples of these are *neighbor* relations between image objects that share a common border and *part_of* relations between indirectly and directly mapped objects.

Both image objects and relations can be associated with numeric and nominal attributes. For image objects, such attributes typically store features like their size, shape, intensity statistics, or classification labels. For relations, they store inherent parameters, like the border length between adjacent objects (see Figure 1).

2.1 Persistence and Query

For real world applications, it is essential to have means of making the data model persistent and to query subsets of its content. Attributed relational graphs can naturally be expressed in terms of the relational data model (Chang and Fu, 1979)—the foundation of all relational databases. Relational databases have the favorable properties of being well established, scalable beyond the limits of working memory and to enable fast and complex queries on the basis of SQL. On this account, the presented concept relies on relational databases for data management.

Since it turned out to be difficult to devise a general database schema for a broad range of applications, the database layout is left to be adapted specifically to the application. However, for reasons of interoperability and clarity, the database layout is expected to comply to the following conventions. The presented concept distinguishes between three types of database tables for objects, relations and attributes.

Object tables store the identifiers of image objects with a common mapping. Accordingly, object tables contain only a single integer column “id”. Directly mapped image objects, that is, objects whose identifiers correspond to pixels values in the labeled image, are stored in a special “base_objects” table. Indirectly mapped image objects are stored together when they belong together semantically.

Relation tables store a common type of relations between image objects. They contain two integer columns “src” and “dest” that store the identifiers of the source and destination objects joined by one relation. For undirected relations, the table contains two entries in opposite directions.

Attribute tables store the attributes of image objects and relations. For objects attributes, the first column “id” contains the identifier of the image object. For relation attributes, the first two columns “src” and “dest” contain the identifiers of the connected objects. In addition to these key columns, attribute tables can have an arbitrary number of integral, real or string columns that store the values of the actual attributes. While it would be possible to have one attribute table to store the attributes of all objects, it is generally advised to provide one table per extraction method. Since most attributes are only computed for subsets of objects, the distributed storage of attributes reduces redundancy and, in turn, the size if the database and computational costs of queries.

2.2 Reasoning

Typically, image analysis requires knowledge on the objects in an image like their size, shape, intensity characteristics, or spatial distribution patterns. The computation of such information is commonly called “feature extraction” while the evaluation of objects on the basis of such features is commonly called “classification”.

The first step in every object-based analysis method is to derive an initial set of objects. This requires a segmentation of the original image that delineates the basic semantic entities. For this purpose, common methods like the watershed transformation (Vincent and Soille, 1991) are feasible. However, depending on the domain, customized segmentation methods might also be considered.

After the initial set of objects has been inserted into the database, all further steps complement the database with additional attributes, objects or relations. In the beginning, the calculation of features and relations helps to group base objects that represent parts of bigger structures. By merging these objects, more complex structures can be identified, thus gaining more contextual information. By extracting spectral or structural features for the merged objects, further classifications are possible on which, again, feature extraction and classification can be performed. That way, the object-based analysis of images be-

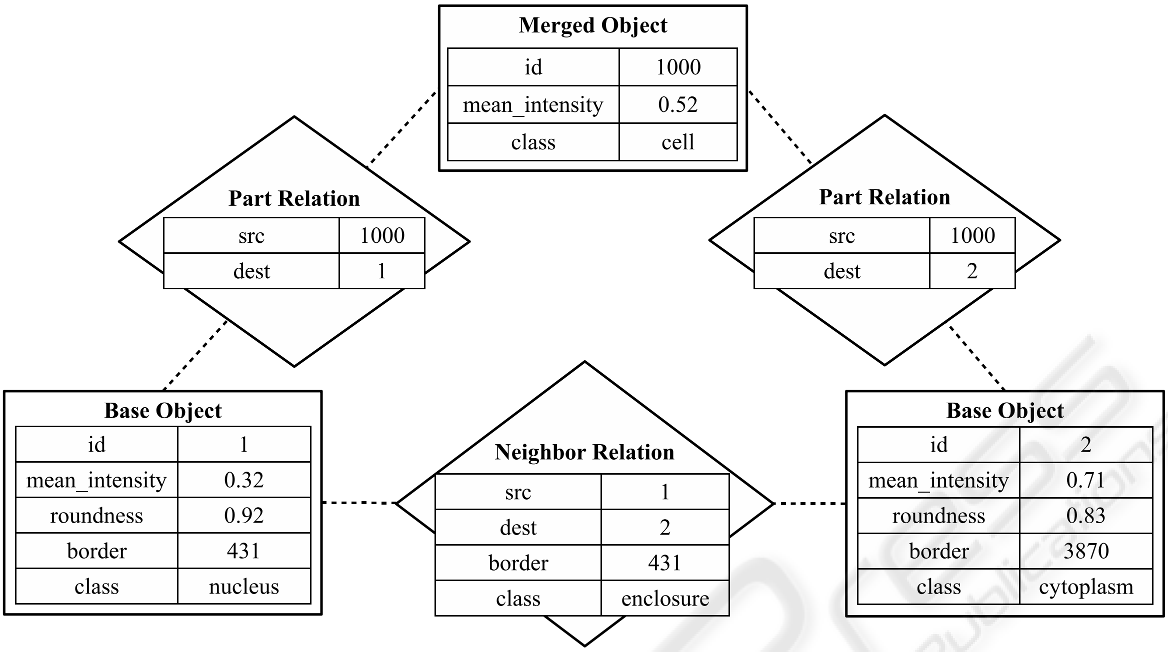

Figure 1: Image representation through an attributed relational graph. Image objects are depicted as rectangles, relations are depicted as diamonds. Both image objects and relations are associated with numeric and nominal attributes which store identifiers, feature values or classification labels. The example graph represents two adjacent image objects—classified as "nucleus" and "cytoplasm"—which constitute one "cell" object.

comes an iterative process of information extraction and classification.

Since every step builds on information gained in the previous step, the object-based analysis of images results in a reasoning process that ends when the structures of interest have been identified. It is during this process, where domain knowledge should drive the selection of meaningful features and classification schemes.

3 REFERENCE IMPLEMENTATION

For practical evaluations, we created a reference implementation of the proposed concept in form of a C++ programming library. While the methodology is independent of any software platform, we integrated the library into the MeVisLab platform (MeVisLab, 2010) in order to take advantage of its comprehensive pixel-based image processing and prototyping capabilities. The general design of the library follows a procedural style with every basic analysis operation being implemented as one function. In addition, a special Selection class was conceived which enables the symbolic definition of subsets of objects or relations. Data management and persistence are accomplished by incorporating the embedded SQLite database—a popular open-source software, that is very efficient in terms of memory requirements and speed.

Since the extraction of meaningful features plays a crucial role in object-based image analysis, the reference implementation provides a fundamental set of feature extraction algorithms. For characterizing the intensities within objects, it enables the extraction of intensity statistics and texture features like local binary patterns (Ojala et al., 2002). For characterizing structural properties of objects, it enables the extraction of shape features like the volume, surface or orientation. Besides features that relate to objects, the reference implementation also includes algorithms for the extraction of relation features, such as the common border area between two objects.

During the reasoning process, domain knowledge is applied via the classification of objects or relations in dependence of their features. If the knowledge can be stated in terms of attribute conditions, the reference implementation enables the explicit classification of objects or relations. If the knowledge cannot be expressed explicitly, machine learning methods can be utilized, like the K-Nearest Neighbors, the Naive Bayes, and the Random Forests classifier (Jain et al., 2000; Breiman, 2001). Knowledge that concerns the spatial context of objects, can be applied by merging related objects to form new indirectly-mapped composite objects.

While this basic set of feature extraction and classification methods provides a good foundation for object-based image analysis, it cannot meet all demands of any image type and problem. Therefore, the reference implementation is easily extensible with additional algorithms in accordance to custom requirements.

An important aspect of object-based image analysis is the visualization of image objects in the context of the image. For this, the application must be capable of rapidly determining all objects within the current field of view. While this can be naively performed by intersection testing with all objects in the database, this becomes unfeasible with a large number of objects. Therefore, the reference implementation stores the bounds attributes of base objects in a special R-Tree data structure (Guttman, 1984) which enables fast range queries. Fortunately, the SQLite database is capable of representing R-Trees as virtual tables, so that the storage of bounds attributes remains consistent to the relational data model. In this manner, the reference implementation provides means for highlighting the borders of image objects and for visualizing the respective attributes in a heat-map-like fashion.

4 CONCLUSIONS

We believe that object-based image analysis algorithms must always be tailored specifically to the problem. Therefore, we propose a generic approach that provides the foundation for the management and processing of arbitrary image objects.

The flexibility of the concept is achieved by using the attributed relational graph as the underlying data model. This enables us to represent objects and relations with their features without imposing a specific ontology. Their actual meaning can be adapted to the individual domain of the application.

The use of relational databases for data management has several benefits. First of all, it allows us to use SQL for stating classification rules, thus greatly supporting the reasoning process. In addition, we gain scalability beyond the limits of working memory.

Special emphasis was put on keeping the design clear and simple. Not only does this foster maintainability but also acceptance among users.

The reference implementation is currently employed in two applications from medical and histological image processing that are under development. However, in order to conclusively prove its broad applicability, the proposed concept has to be evaluated upon more problems from different domains.

REFERENCES

Aksoy, S. (2006). Modeling of remote sensing image content using attributed relational graphs. *Lecture Notes in Computer Science*, 4109:475–483.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Chang, N.-S. and Fu, K.-S. (1979). A relational database system for images. In *Pictorial Information Systems*, pages 288–321. Springer.

Guttman, A. (1984). R-trees: A dynamic index structure for spatial searching. In *Proceedings of the 1984 ACM SIGMOD international conference on Management of data*, pages 47–57. ACM New York, NY, USA.

Hay, G. J. and Castilla, G. (2006). Object-based image analysis: strengths, weaknesses, opportunities and threats (swot). In Lang, S., Blaschke, T., and Schöpfer, E., editors, *1st International Conference on Object-based Image Analysis (OBIA 2006)*.

Jain, A., Duin, R., and Mao, J. (2000). Statistical pattern recognition: A review. *IEEE Transactions on pattern analysis and machine intelligence*, pages 4–37.

MeVisLab (2010). Medical image processing and visualization. http://www.mevislab.de. Retrieved February 24, 2010.

Ojala, T., Pietikäinen, M., and Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 971–987.

Schäpe, A., Urbani, M., Leiderer, R., and Athelogou, M. (2003). Fraktal hierarchische, prozeß-und objektbasierte Bildanalyse. *Procs BVM*, pages 206–210.

Shackelford, A. and Davis, C. (2003). A combined fuzzy pixel-based and object-based approach for classification of high-resolution multispectral data over urban areas. *IEEE Transactions on GeoScience and Remote sensing*, 41(10):2354–2363.

Vincent, L. and Soille, P. (1991). Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE transactions on pattern analysis and machine intelligence*, 13(6):583–598.