Chapter 1.2

Progressing from object-based to object-oriented image analysis

M. Baatz [1], C. Hoffmann[1], G. Willhauck [1]

[1] Definiens AG, Munich, Germany;
{mbaatz, choffmann, gwillhauck}@definiens.com

KEYWORDS: Segmentation, object-based image analysis, object-oriented image analysis, region-specific segmentation, feature extraction.

ABSTRACT: This research describes an advanced workflow of an object-based image analysis approach. In comparison to the existing two-staged workflow where typically a segmentation step is followed by a classification step, a new workflow is illustrated where the objects themselves are altered constantly in order to move from object primitives in an early stage towards objects of interest in a final stage of the analysis. Consequently, this workflow can be called "object-oriented," due to the fact that the objects are not only used as information carriers but are modelled with the continuous extraction and accumulation of expert knowledge. For better demonstration, an existing study on single tree detection using laser scanning data is exploited to demonstrate the theoretical approach in an authentic environment.

1 Introduction

Recent developments in remote sensing made it possible to obtain data of a very high spatial resolution which allows extraction, evaluation, and monitoring of a broad range of possible target features. At the same time, the demand to automate image analysis in operational environments is constantly growing. However, the variety and number of different features to be extracted, add challenges specifically in terms of modelling and auto-adaptive procedures.

The advantage of a spatial resolution with pixel sizes significantly smaller than the average size of the object of interest comes with the disadvantage of an abundance of spatial detail and the accordingly huge amount of data to be processed. To overcome this drawback, the object-based image analysis approach has proven to be an alternative to the pixel-based image analysis and a large number of publications suggest that better results can be expected (Baatz and Schäpe 2000, Willhauck et al. 2000, Hay et al. 2005, Kamagata et al. 2005, Manakos et al. 2000, Whiteside et al.2005, Yan et al. 2006).

The object-based approach suggests a two-staged approach. In the first step pixels are merged to object clusters, possibly in a multi-level object hierarchy, which then will be analysed and classified in the second step. This means that, the created objects influence the classification result to a large extent although they might not represent the final objects of interest (i.e. single buildings, trees, etc.) already. Because the objects remain unchanged once they are created, and subsequently serve as basis for the actual analysis, this workflow can be called “object-based image analysis”. A successful object-based image analysis results in the correct labelling / classification of regions rather than extracting final objects of interest for instance like trees, acres, buildings or roads in their final shape.

In comparison to the “object-based” workflow, this paper describes an alternative, more advanced workflow which not only uses object clusters as the basis for a classification analysis but brings the objects themselves and the shaping of the objects in the focus of the analysis.

This alternative workflow starts with creating object clusters and aims to produce desired objects of interest with correct shape and correct labelling. Why is this required? The accuracy and the significance of the final measurements, numbers, and statistics directly and actually critically depend on the quality of segmentation. Relevant information such as numbers, shapes or other statistics per unit is only accessible if trees are not only correctly labelled as “tree area” but also are correctly extracted tree by tree as “tree objects” or ”tree units”.

Typically, the correct extraction and shaping of objects of interest requires more advanced models, domain knowledge and semantics, in order to cope with the specific characteristics of the structure and to sort out ambiguities that often occur. The more or less simple and knowledge-free segmentation procedures used to produce object clusters or object primitives almost never succeeds in extracting objects of interest in a robust and reliable manner. Furthermore, different types of target objects also need different strategies for their extraction. In order to support this, decisions need to be made throughout the process that classifies different regions and thus make them accessible to different extraction approaches.

2 Methodology

2.1 Object-oriented image analysis workflow

The workflow described here starts with object primitives as well. However, in contrast to the object-based workflow, it uses these objects not only as information carriers but also as building blocks for any further shape modification, merging, or segmentation procedures. In a whole sequence of processing steps, where segmentation steps alternate with evaluation and classification steps, these object primitives are constantly altered until they become the desired objects of interest. Because this strategy aims for correct shaping and classification of objects, and objects are also used at every step during the procedure as the central processing unit, serving both as information provider and as building blocks, this workflow consequently can be called "object-oriented image analysis".

This approach can be realised by using Cognition Network Technology (CNT), distributed by Definiens AG, with its included modular programming language CNL (Cognition Network Language). Amongst typical programming tasks like branching, looping and variable definition, CNL enables to build and perform specific analysis tasks based on hierarchical networks of objects and essentially supports an object-oriented image analysis workflow.

This workflow can be described best with a spiral and is illustrated in Fig. 1. The entire process is iterative and starts in the first step with the creation of object primitives using any (knowledge-free) segmentation algorithm. The next step uses the object primitives in order to perform a first evaluation and classification, thus introducing semantics. Building on this result, the subsequent step allows refinement or improvement of the segmentation locally for a specific class. Thus, the whole process alternates iteratively between local object modification on the one hand and local object evaluation and classification on the other. By using such an iterative approach, different object classes can be addressed with different object modification strategies.

During this process, the objects are altered from stage to stage until they represent the final target objects. Only the combination of correct shaping and correct classification characterizes the target objects; otherwise the results will be insufficient. The spiral in Figure 1 represents this alternation between segmentation and classification in the object-oriented approach.

As the analysis progresses, classification detail and accuracy are growing together with segmentation detail and accuracy. Whereas the process starts with rather simple and knowledge-free segmentation steps, more and more expert and domain knowledge is introduced and used in later steps. The more closely the process approximates the final target objects, the higher is the abstraction from the original image information.

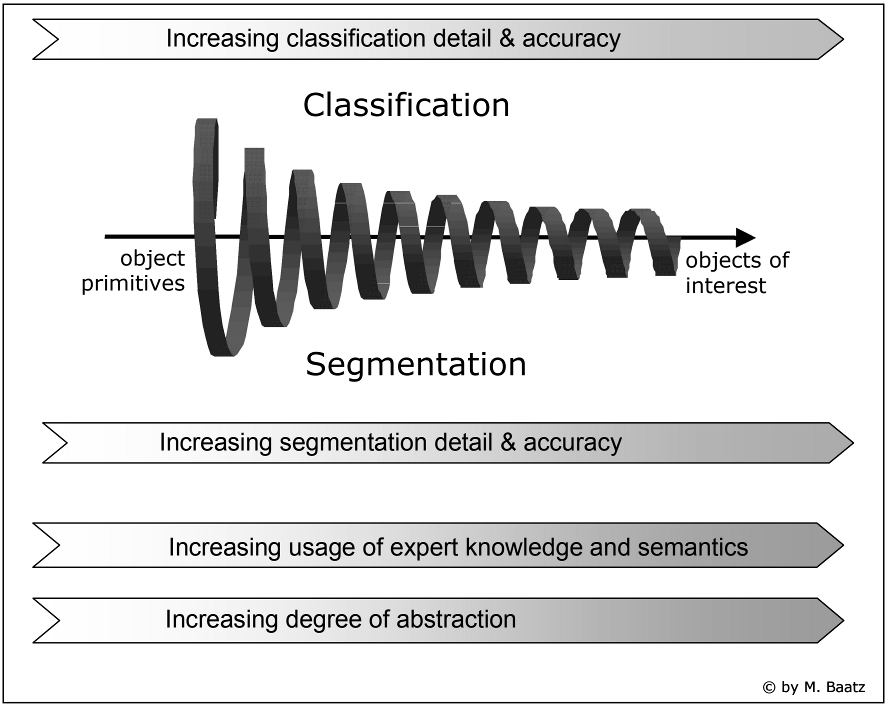

Fig. 1. Object-oriented image analysis: the generic procedure

2.2 The Object Domain

A central concept in the object-oriented workflow is the Object Domain. It defines for each algorithm the specific subset of objects to which the algorithm—independent if segmentation or classification—will be applied. It therefore enables implementation of locally specific strategies.

The Object Domain characterizes a subset of objects in the object hierarchy through the hierarchical object level, the classification of objects and/or specific attributes of objects.

The concept can be further differentiated. Starting with objects of a specific Object Domain, subdomains can be defined operating over the networked neighbourhood. In a hierarchical object network, subdomains of an object can for instance be neighbour objects, subobjects or a superobject in a defined distance, with a specific classification and/or with specific attributes.

In the continuously alternating object-oriented image analysis workflow, the Object Domain is the essential link between segmentation and classification.

2.3 Evaluation and classification aspects

During the course of the image analysis, localized evaluation and classification is essential. Before objects can be distinguished into different types by classification, object attributes must be evaluated. These attributes can be intrinsic to the object—such as shape, size or spectral characteristics—or they can be derived from operations over the networked context of an object. Contrasts, embedding, relative location, and composition are good examples.

Independent of the individual classification method it always can be applied to a specific subset of objects defined through an Object Domain. Thus very specific measurements and decisions can be performed locally.

If the evaluation is more complex, intermediate results can be stored in variables. Whereas object variables hold information specific to an individual object global variables hold information specific to the whole scene.

Making decisions that relate to measurements stored in variables is an important tool for auto-adaptive strategies which play a crucial role in making solutions robust over any expected variability of object characteristics. As mentioned before, the attributes derived from objects and used for classification critically depend on the way the object has been processed.

2.4 Segmentation aspects

The term “Segmentation” is used here as the summary of all procedures that build, modify, grow, merge, cut or shrink objects. In principal, segmentation techniques can be distinguished depending on if they are used to produce initial object primitives starting with pixels (A) or if they further process already existing objects (B). Typically, an object oriented image analysis process starts with (A) and performs all further object modification with (B).

Independent of the specific algorithm the modification (B) can be applied to a specific subset of objects defined through an Object Domain. Thus locally very specific modifications can be performed.

A number of segmentation methods are working with seed objects as a starting point and candidate objects in the networked neighbourhood which are used to modify (merge, grow, shrink, cut) the seed unit. In these cases, both types – seeds and candidates – can be defined through different Object Domains in order to constrain the algorithm and make it more selective.

2.5 Fractal structure of the workflow

The described workflow of alternating segmentation and classification steps is inherently fractal.

In many cases, a specific segmentation step needs preparation in form of an evaluation and a classification of objects which shall be modified or which contribute to a specific modification. A good example for this is the object-oriented watershed algorithm described in the case study below.

Symmetrically, image objects are not always directly in the appropriate state to provide the relevant information needed to do a certain decision. In theses cases preparation is needed through an adequate segmentation of objects. For instance, if the size of water bodies matters and at the current state of processing water bodies are represented through correctly classified object primitives then merging those primitives into complete water body units allows to access the needed information.

Fig. 2 illustrates how subprocesses themselves show the alternation of segmentation and classification.

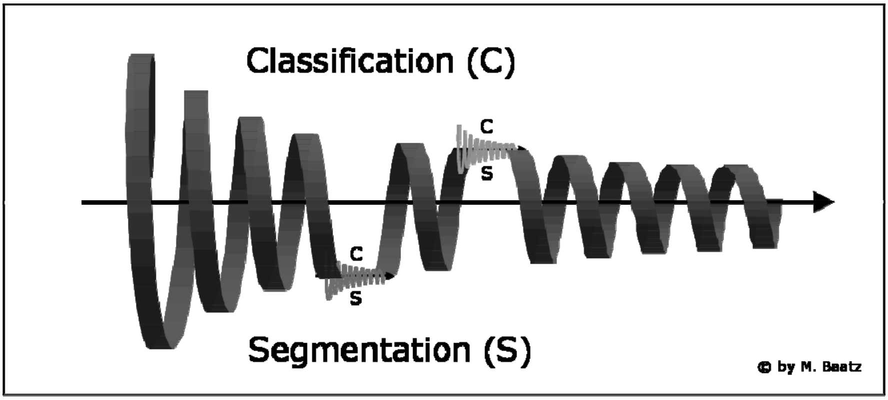

Fig. 2. Fractal approach of the object-oriented image analysis workflow

2.6 Modular Structure

Typically, an overall analysis procedure consists of a number of sub modules that each addresses a certain target class. Each individual module is an encapsulated object-oriented image analysis task. Fig. 3 shows an example for a sequential modular approach.

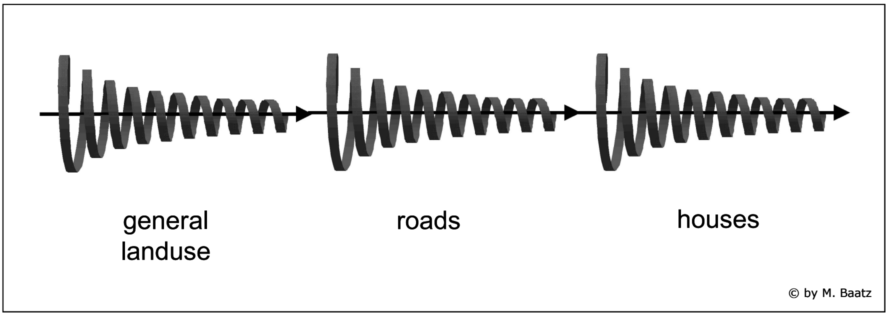

Fig. 3. Modular Structure of the object-oriented image analysis workflow

3 Case study – single tree detection

Individual object-based tree detection has been discussed in a number of publications as well as the use of the local maxima approach (Pitkänen et al. 2001, Pitkänen et al. 2004, Tiede et al. 2004, Tiede et al. 2005).

The following case study exemplarily demonstrates an object-oriented image analysis workflow. It is a solution for single tree detection using airborne laser scanning data and was carried out by Tiede and Hoffmann (2006).

Starting with object primitives the objects are constantly evaluated and altered until the target objects in form of tree units are found. The approach used can be called an object-oriented and knowledge-based watershed algorithm.

Fig. 4 gives an overview over the workflow as described in detail in the following chapters.

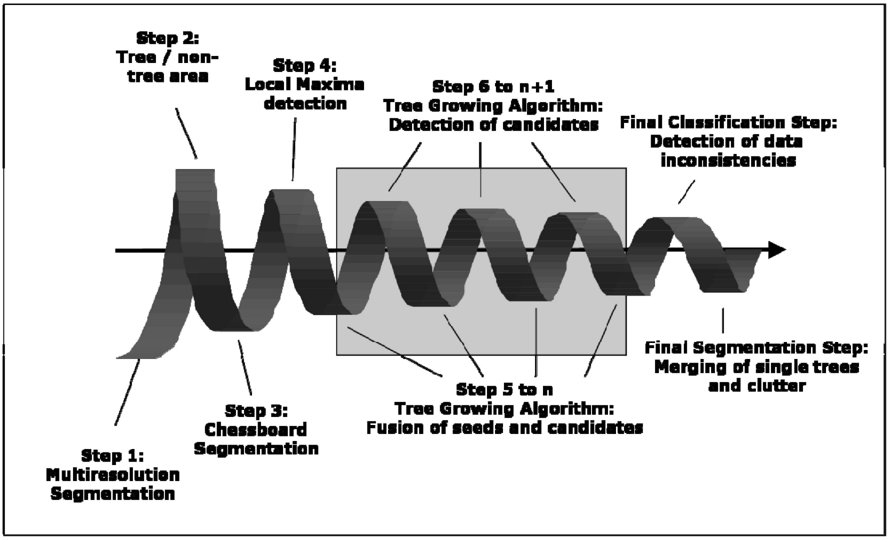

Fig. 4. Single tree detection workflow in the context of object-oriented image analysis. The tree growing algorithm consists of a number of iterations itself

3.1 Used data

Fig. 5 shows the used data derived from a digital surface model (DSM) of an airborne laser scanning dataset with a ground resolution of 0.5 meters.

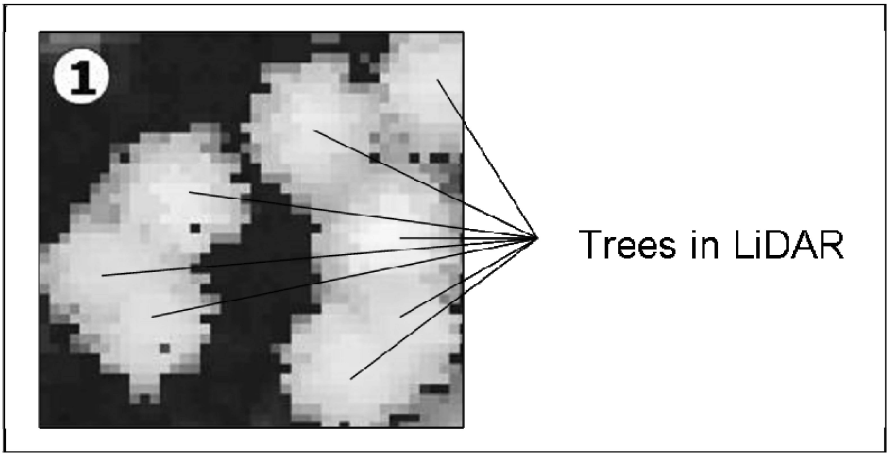

Fig. 5. Crown Model from airborne laser scanning. In this small subset, eight trees can be seen represented by high (bright) values. Tiede & Hoffmann (2006), edited

3.2 Tree / non-tree area

The first step is to distinguish the background from the tree area. It turns out that object primitives best suited to provide information for this decision can be created with Multiresolution Segmentation. Theses primitives might each cover a number of trees but they are small enough to reliably separate tree area from background.

The subsequent classification step distinguishes areas which are elevated and, thus, potentially represent tree crowns from the background. All further processing steps will build on this basic decision and use the tree area as the Object Domain.

Fig. 6. Step 1: Multiresolution Segmentation and Classification of tree / non tree area. Tiede and Hoffmann (2006), edited

3.3 Local maxima

The model assumes that using a digital surface model a tree can be extracted starting with its tree top working as a seed point.

The currently available object primitives are by far too large to support this analysis. In order to achieve the needed fine granularity all objects in the Object Domain “tree area” are cut down to single pixels using a Chessboard segmentation. Thus, the region “tree area” remains unchanged; however, it now consists of small pixel-sized object primitives.

Now all object primitives that represent a local elevation maximum are classified within the tree area. A classification step operating on the Object Domain “tree area” compares the elevation of each object with that of its neighbours in a certain distance and if it is higher than all others it classifies the object as local maximum (Fig. 7).

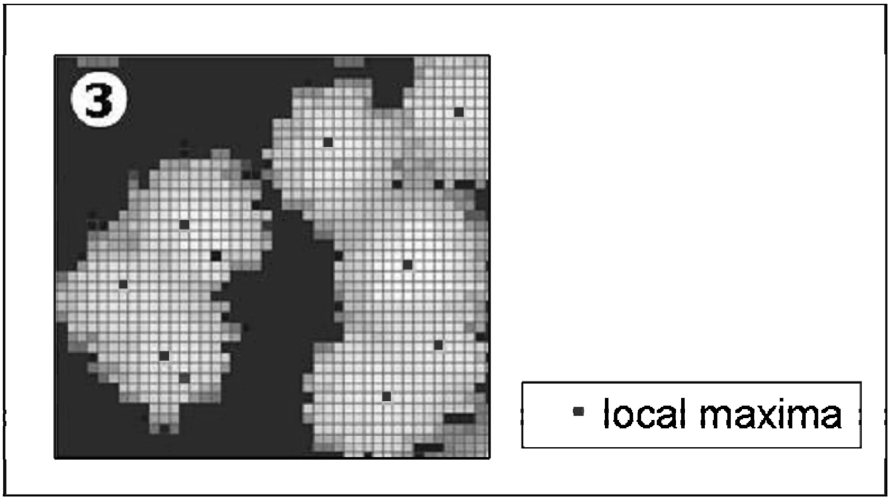

Fig. 7. Step 2: Domain-based break-down of tree area into pixel-sized objects and application of local maxima algorithm. Each maximum represents the highest point of a single tree. Tiede and Hoffmann (2006), edited

3.4 Tree building algorithm

Fig. 8 shows one step within the growing process of the single trees. This growing process is done by a simultaneous merging procedure of the neighbour objects of the already existing tree tops which work as “seeds”.

Each tree is grown by using contextual information (similar height) to decide whether the neighbourhood objects are belonging to the currently growing tree. Stop criteria are used to prevent the trees from growing excessively or in an unwanted direction. Since the growing of the trees is done for all trees simultaneously, the trees can only grow until they reach another tree.

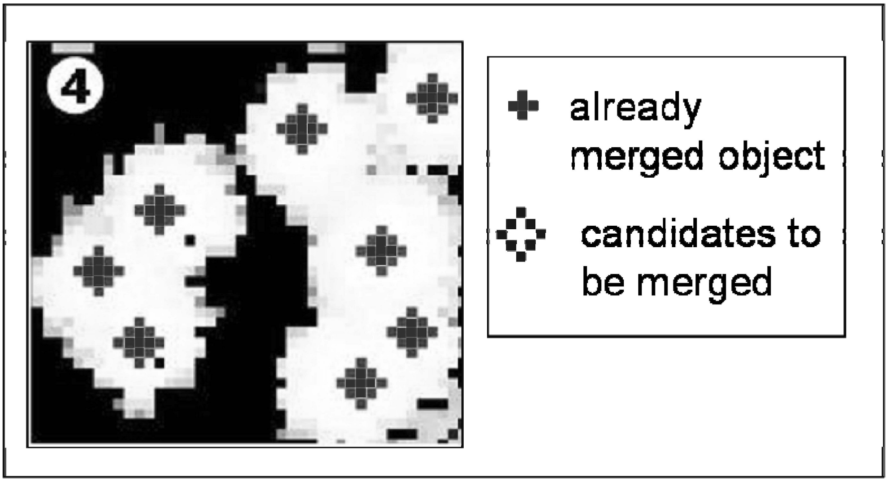

Fig. 8. Step 3: Growing algorithm around tree seeds by similarity. If surrounding objects are matching, they are merged. The growing algorithm itself consists of iterations until the final tree objects are found. Tiede and Hoffmann (2006), edited

The tree growing loop is an example for a sub-procedure which as well can be thought of as a spiral. In each iteration of the loop, there is a classification and a merging step. First, all appropriate neighbour objects are classified as candidates whom are going to be merged with the current tree object. In the second step, these classified objects are then merged with the currently processed tree object to form one large tree object.

Since the number of iterations is defined by a stop criteria and the growing of other trees, the number of iterations is not fixed. It will continue as long as candidate objects exist which can be merged with the growing tree objects.

3.5 Result of the growing procedure

After the tree growing loop, single tree objects exist which are almost representing the desired objects of interest. Fig. 9 shows the result of the tree growing loop. Because of data inconsistency of the laser scanning data, individual pixels can be found with no data or values that are not fitting. These pixels can be removed in a last refinement step, where context information is used again to remove these data errors.

It is important to approximate the tree objects as good as possible to the shape of naturally looking trees because in a potential analysis of the trees in a later stage, parameters like the tree crown diameter might be important for any analysis.

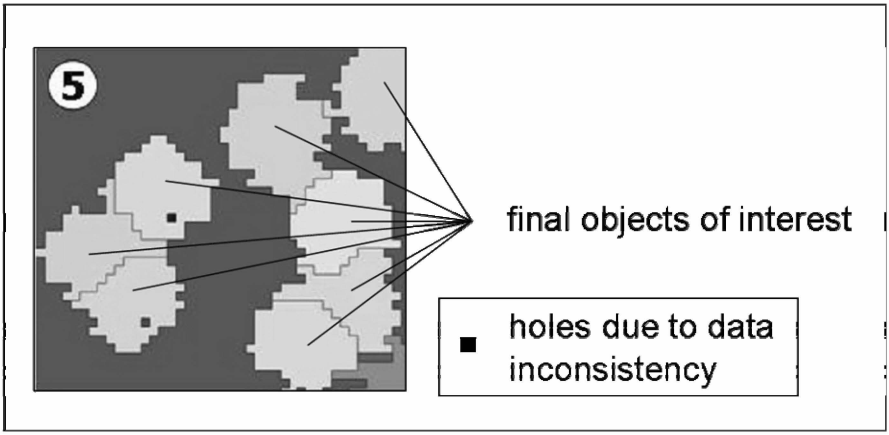

Fig. 9. Result after the tree growing algorithm is finished. Holes due to data inconsistency are still existing. Tiede and Hoffmann (2006), edited

3.6 Remove data inconsistencies

Fig. 10 shows the result after removing data inconsistencies. According to the theoretical workflow shown in Fig. 1, at the end of the alternating segmentation and classification steps, the final objects of interest are modelled in terms of their shape as well as their classification.

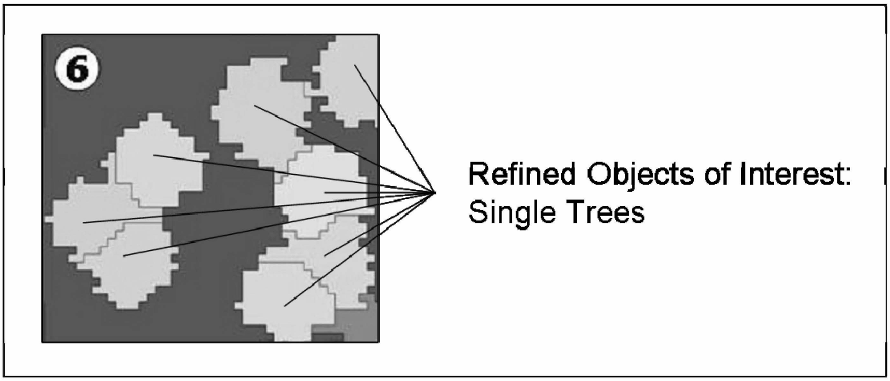

Fig. 10. Final result for single tree detection. Holes were removed by the use of context information. Tiede and Hoffmann (2006), edited

4 Discussion

Considering the complexity and often ambiguities of image analysis problems it is clear, that in most cases more complex models and semantics are needed. The spiral process turns out to be a very efficient paradigm for supporting more complex analysis problems which include modelling, semantics or deal with ambiguities. The object-oriented workflow has proven in many applications (both in Earth and Life Sciences) to be useful to extract objects of interest in an automated manner which is not supported by the object-based approach. Not only if objects of interest are to be extracted but also if only a correct labelling of regions is requested a pure object-based approach is often limited.

In the spiral which defines the object-oriented approach, each step builds on the results of the previous. There is a mutual dependency between segmentation and classification: the attributes and the quality of attributes used to evaluate and classify image objects directly depend on the objects and how they were formed before. The precise formation of objects on the other hand needs specific semantics, models or knowledge how to sort out ambiguities. Since local classification and local segmentation interact and mutually depend on each other in manifold ways through this process, they can be described in a colloquial sense as the Yin and Yang of object-oriented image analysis.

In other words, the object-oriented image analysis workflow overcomes a commonly known problem which can be called the "hen and egg problem" in image segmentation: for a successful classification of a certain feature class, the object-primitives need to exist already in a form at least very near, sometimes even identical to the final target objects. On the other hand, in order to achieve such objects, local evaluation and semantics provided by classification are needed during the segmentation process itself. For that reason, the workflow suggests a step-wise approximation from object primitives with a coarser classification and shape in an early stage to objects of interest and the according detailed classification in the final stage.

5 References

Baatz M and Schäpe A (2000) Multiresolution Segmentation – an optimization approach for high quality multi-scale image segmentation. In: Angewandte Geographische Informationsverarbeitung XII, Strobl J, Blaschke T, Griesebner G (eds), Wichmann, Heidelberg, Germany, pp 12-23.

Hay GJ, Castilla G, Wulder MA, Ruiz JR (2005) An automated object-based approach for the multiscale image segmentation of forest scenes. In: International Journal of Applied Earth Observation and Geoinformation, Vol. 7, pp 339-359.

Kamagata N, Akamatsu Y, Mori M, Qing Li Y, Hoshinoy Y, Hara K (2005) Comparison of pixel-based and object-based classifications of high resolution satellite data in urban fringe areas. In: Proceedings of the 26th Asian Conference on Remote Sensing. Hanoi, Vietnam. 7–11 November 2005.

Manakos I, Schneider T, Ammer U (2000) A comparison between the ISODATA and the eCognition classification methods on basis of field data . In: ISPRS, Vol. XXXIII, Amsterdam, 2000.

Pitkänen J, Maltamo M, Hyyppä J, Wei Yu X (2004) Adaptive methods for individual tree detection on airborne laser based canopy height model. In: The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXVI-8/W2, Freiburg, pp 187-191.

Pitkänen J (2001) Individual tree detection in digital aerial images by combining locally adaptive binarization and local maximal methods. Canadian Journal of Forest Research, Vol. 31, pp 832-844.

Tiede D, Burnett C, Heurich M (2004) Objekt-basierte Analyse von Laserscanner- und Multispektraldaten zur Einzelbaumdelinierung im Nationalpark Bayerischer Wald. In: Strobl S, Blaschke T, Griesebner G (eds), Angewandte Geoinformatik 2004, Wichmann, Heidelberg, Germany, pp 690–695.

Tiede D, Hochleitner G, Blaschke T (2005) A full GIS-based workflow for tree identification and delineation using laser scanning. In: The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVI, Part 3/W24, Vienna, pp 9 – 14.

Tiede D, Hoffmann C (2006) Process oriented object-based algorithms for single tree detection using laser scanning data. In: EARSeL-Proceedings of the Workshop on 3D Remote Sensing in Forestry, 14th-15th Feb 2006, Vienna, pp 151-156.

Whiteside T, Ahmad W (2005) A comparison of object-oriented and pixel-based classification methods for mapping land cover in northern Australia. Proceedings 2005. Melbourne: Spatial Sciences Institute. ISBN 0-9581366-2-9.

Willhauck G, Schneider T, De Kok, R Ammer U (2000) Comparison of object oriented classification techniques and standard image analysis for the use of change detection between SPOT multispectral satellite images and aerial photos . In: ISPRS, Vol. XXXIII, Amsterdam, 2000.

Yan G, Mas JF, Maathuis BHP, Xiangmin Z, Van Dijk PM (2006) Comparison of pixel-based and object-oriented image classification approaches–a case study in a coal fire area, Wuda, Inner Mongolia, China. International Journal of Remote Sensing. Volume 27, Number 18 / 20 September 2006, 4039–4055.

Springer

http://www.springer.com/978-3-540-77057-2

Object-Based Image Analysis
Spatial Concepts for Knowledge-Driven Remote Sensing Applications
Blaschke, Th.; Lang, S.; Hay, G. (Eds.)
2008, XVII, 817 p. With online files/update., Hardcover
ISBN: 978-3-540-77057-2