Institut für Geodäsie und Geoinformation
Professur für Protogrammetrie

Interpretation of Aerial Images with Learned Graphical Models

Dissertation
zur
Erlangung des Grades
Doktor-Ingenieur
(Dr.-Ing.)
der
Landwirtschaftlichen Fakultät
der
Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt am 29.06.2015 von
Hanns-Florian Schuster
aus Mülheim an der Ruhr

Referent:	Prof. Dr.-Ing. Dr. h.c. mult. Wolfgang Förstner
Korreferent:	Prof. Dr. Cyrill Stachniss
Tag der mündlichen Prüfung:	23.10.2015
Erscheinungsjahr:	2016

Zusammenfassung

Die vorliegende Arbeit präsentiert einen neuen Ansatz der Bildinterpretation von Luftbildern mit Hilfe von gelernten Bayes-Netzen. Luftbilder sind der Ausgangspunkt zur Herstellung von Karten und damit der Ausgangspunkt für Infrastrukturplanungen und Navigationsdaten.

Das Verfahren arbeitet auf einem Regionen-basierten hierarchischen Merkmals-Nachbarschafts-Graph. Dieser enthält alle aus dem Bild extrahierten homogenen Regionen, inklusive ihrer Nachbarschaftsstrukturen. Die Regionen werden dabei durch 17 Bild-Merkmale beschrieben. Diese sind z.B. Farbe, Struktur, Form oder Symmetrien. Die Nachbarschaftsbeziehungen selbst werden durch sieben Merkmale attributiert.

Das Bayes-Netz besteht aus Knoten für die Beobachtung der Regionen, ihrer Merkmale und Nachbarschaften, sowie die Aggregationsstufen der Cliquen, Objekte und der Szene als Ganzes. Die Regionen des Merkmal-Nachbarschafts-Graphen sowie ihre Attribute werden als Beobachtungen in den entsprechenden Knoten des Bayes-Netzes eingeführt. Im Lern-Schritt wird dazu auch der Typ der Bildszene als Beobachtung eingeführt. Die anderen Knoten des Bayes-Netzes sind unbeobachtet. Das präsentierte Verfahren der Bildinterpretation ist zweistufig: In der ersten Stufe wird das Bayes-Netz anhand von vorhandenen Interpretationsergebnissen auf bekannten Bildern trainiert. Dabei werden die Struktur des Bayes-Netzes und die Wahrscheinlichkeitsdichten gelernt. Die gelernten Parameter für die Abhängigkeiten und die Wahrscheinlichkeitsdichten, die das Bayes-Netz repräsentiert, sind das Ergebnis der ersten Stufe. Sie werden abgespeichert und in der zweiten Stufe verwendet.

Die zweite Stufe benutzt die Parameter, welche in der ersten Stufe ermittelt wurden, zur Interpretation. Dafür werden wiederum die Elemente aus dem Merkmals-Nachbarschafts-Graph als Beobachtung für das Bayes-Netz benutzt. Dann wird in einer iterativen Maximum-a-posteriori Schätzung die bestanpassende Struktur als Lösung für das Bayes-Netz gesucht. Die Zustände der Knoten im Bayes netz repräsentieren nun die Interpretation des Bildes mit den im ersten Schritt erzeugten Vokabeln. Ergebnisse der so gefundenen Interpretation werden visualisiert und ausgewertet.

In verschiedenen Experimenten wird die Stabilität und Robustheit des Verfahrens auf vier verschiedenen Datensätzen von Luftbildbefliegungen gezeigt.

Abstract

In this thesis we present a new approach for image interpretation of aerial images using learned graphical models.

The approach uses a region based hierarchical feature adjacency graph. This contains homogeneous regions that were extracted out of the image as well as its neighbor relationships. For each region there are 17 image features extracted to describe the region, e.g. color, structure and symmetries. The neighbor relations are attributed by seven features describing their geometrical relation. The Bayes net consists of nodes for regions and their image features as well as the neighbor relationships. It also models the higher aggregated elements with nodes for cliques, objects and the image scene. The regions of this adjacency graph and the describing features are used as observations for the nodes of the Bayes net. In the first learning step also the scene node is introduced as observation the other nodes are hidden i.e. not observed.

The presented approach has two stages. In the first stage the Bayes net is trained with known ground truth data. By introducing the observations, a structural learning algorithm searches the best net structure and learns the probability distributions and the dependencies of the nodes of the Bayes net. The learned parameters are the result of the first stage. They are saved and used in the second stage

The second stage interprets new images using a Bayes net with the parameters of the first stage. Again, the regions and features of the region based feature adjacency graph are introduced as observations. Using an iterative maximum a posteriori estimation, we search for the optimal Bayes net structure to describe the underlying image. The states of the nodes of the Bayes net represent now the interpretation according to our learned vocabulary.

Contents

Chapter 1

Introduction

In this thesis we will present an approach for interpreting the content of aerial images. We develop a statistical model that is capable to classify and interpret image regions in a hierarchical way and create a scene interpretation of it. The parameters of the interpretation process represent the knowledge about the scene and its objects. These parameters are determined in a preceding learning step from annotated images (Fig. 1.3). Although the learning and interpretation is capable of processing any image content in general, we will focus on the specialties of the interpretation of aerial images.

1.1 Motivation and goal

The detection of objects is a challenging task in computer vision that attracted much attention in the last years. The demand for object detection algorithms today is big. Images are used everywhere since cheap digital camcorders and consumer cameras are available. The quality and size of the images increase and storage is cheap, so there are a lot of huge image databases. Be it the private collection of the last holiday or a professional database or images of an observation camera, to search the database for a certain motive of object in the scene quickly gets painful if there is no label that can be searched easily. Also in applications that apply for machine vision like robotics underline the need for object detection once more.

Due to the historical connection to the cadastre and surveying, photogrammetry had always a strong focus on building and terrain reconstruction which are e.g. essential for mapping.

Also in the classical domain of photogrammetry the demand for aerial images and interpretation is big and not restricted to the geodetic community as the success of

applications like Google Earth show. The use of such data is quickly growing, where nowadays the home computer technology overcomes the difficulty of large images. A main task for photogrammetry is creating maps from aerial photos. The need for the task of mapping can be imagined by looking on mega cities like Mexico City or Bombay. Their daily growth is uncontrollable and fast. Aerial images are in this case the only chance for mapping and decision making. An automatic image interpretation is needed to produce actual maps.

Although the use of additional sensors seem to be useful to supply the recognition process, the presence of additional data is rare and will be expensive for a long time in comparison to the small costs of (digital) images. Also there are tasks like finding bomb shells in the ground 60 years after the Second World War. This is done with the interpretation of old aerial images taken in the 1940s *[Landes NRW 2011]*.

Images provide iconic information that is easily accessible for humans but is not decodable directly for computers. The image feature extraction as part of the low level vision makes the transition from the image as a signal coded by pixels to a symbolic level. But also this level not yet provides the desired information for accessing and interpreting the content of the image. Goal of the detection here is to make a further step to a semantic level which assigns labels to "objects" that are built of features of any kind.

Figure 1.1: Goal of this thesis is to interpret the aerial image on the left side and get an interpretation result like shown on the right side. The colors define the different object classes. In this case there are streets [black], saddle roofs [green], hip roofs [orange], flat roofs [light blue] and other roofs [dark blue] found.

To design an optimal detection method, there are several requirements to be met. It

Task	Existence	Name	Class	Position	Form
Object localisation	+	·	+	?	·
Object reconstruction	+	·	+	+	?
Object identification	·	?	?	+	·
Object detection	?	·	+	·	·
Image interpretation	·	?	?	?	?

Figure 1.2: Tasks in computer vision: +=given; ?=searched; ·=irrelevant, unknown, perhaps searched, perhaps given; Categorization from *[Förstner 2009]*

is helpful if the algorithm is able to learn from examples. This is the form also humans learn and thus this kind of teaching is familiar. The learning phase should be able to learn the probability densities of regions contained in image scenes with few samples and the knowledge base should be expandable for the case we want to add new object classes. The detection itself should be invariant to image transformations and distortions like noise and partial occlusions. It should handle a large number of classes that are part of an ontology specifying a taxonometry and a partonomy and containing spatial or other constraints. Using this, the optimal detection method should detect every object of a scene. Also, the detection should comply in a reasonable short time.

Like shown in Fig. 1.2 there are different kinds of related tasks of dealing with objects in images. These tasks have different input and output and therefore have different complexity to deal with. In object *localisation* and object *reconstruction* the position and form is unknown whereas the existence and the class of the objects are known. In the *identification* there is the definitive knowledge that there is an object on a certain position, which has to be classified. Also the detection needs parameters as input: it determines whether a given object is present in the image or not, possibly reporting its location. In contrast to this, the process of image interpretation normally has no necessary input about the presence of objects in a specific scene. It has only generic knowledge about the objects appearance. Not only the name, class, position and pose is unknown, also the existence of objects has to be proved against the model. It can be seen as object detection with an open list of concurrent objects. To comply with this challenge we need a complex model and an algorithm that is capable to perform these tasks and combine the results.

The goal in this thesis is to develop a method of a scene interpretation of aerial images. The interpretation implements knowledge about the classes that are taught in a learning step. Everything else in the scene will be classified as background. The results of the scene interpretation are instances of objects found in the image scene (see fig. 1.1).

These objects are represented by nodes of a Bayes net. The reprojection of the found object class to the dependent features in the image is displayed in fig. 6.2.

1.2 Strategy for interpretation

Our image interpretation task is composed of two steps. In the first step the method builds up a knowledge base about the objects and their image features, which is the learning step. Then follows the detection and reconstruction in a second step.

The idea that is followed here is that it is easier to let the computer find its model for a certain task than to engineer a set of criteria for the object detection. Therefore the desired objects and their parts are learned in a supervised step. The supervision by the human teacher has not to be provided online, but it can be done by using an annotated image database. The knowledge that is extracted in this step is represented explicitly in parameters. These can be used for classifying new images (fig.1.3).

For carrying out the classification we propose the use of a Bayes net *Koller and Friedman [2009]*. This has certain advantages: The structure can be modeled very flexible and it represents a statistically based scheme. The probabilities and dependencies inside the net are interpretable by humans. Bayes nets are a well understood tool for reasoning and it is possible to learn the probability densities as well as their dependencies. These parameters can be expanded in a second step; a repetition of the learning of the previous data sets after adding new cases is not necessary.

One reason for the choice of this learning approach is the problem of an adequate strength of the model. As shown in *Brunn and Förstner [1995]* and *Kulschewski [1997]* there has to be a consideration between the strength of the model and the strength of the data. This problem can be cleared out by learning the detection model directly from the data. In this case the knowledge is not derived from an engineered model. The problem is to avoid overfitting that would hinder the generalization in learning. Another reason is that there exist a lot of different feature extraction algorithms. Because of their amount and different behavior it is not clear what type of feature is able to contribute evidence to the interpretation. For this reason we provide a bouquet of different feature detectors that are implemented as observations. After the learning procedure, those features, that add evidence to the interpretation process, will have clearly peaked probability distributions. Those which do not serve the interpretation will instead have "flat" distributions and a high variance respectively. The image model enables us to implement features that represent the geometry, topology and color.

The feature extraction part is region based, i.e. the region is in the focus and every

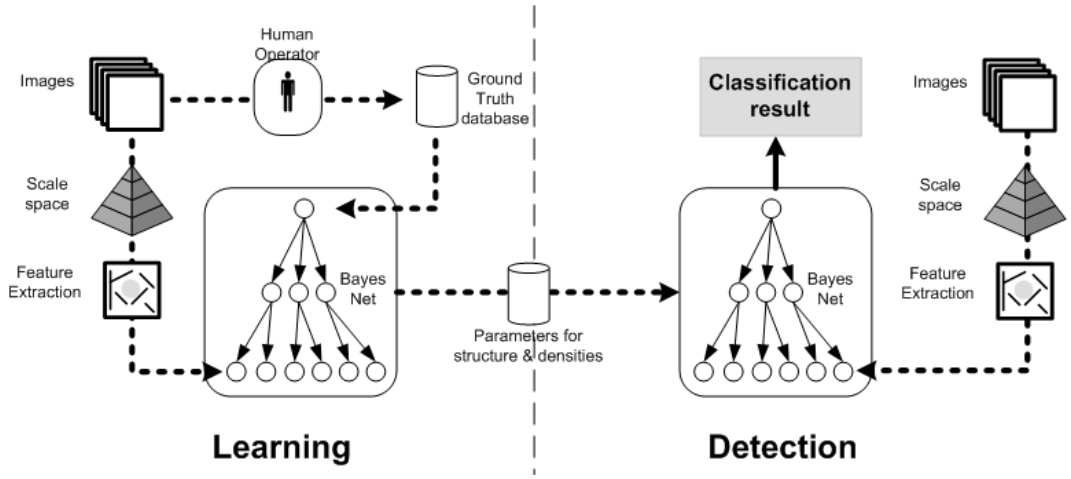


Figure 1.3: The overall strategy: the learning step extracts scene knowledge from images with given ground truth. This knowledge is represented by parameters that define the structure and the probability distributions of the Bayes net. These can be used in the detection step for scene interpretation.

observation is a description of a region: points and lines are incorporated as information about the boundary of regions. This was chosen because the unknown objects are modeled as region like structures and the topology structure is easily accessible in this type of region adjacency graph.

Another possibility is to use tripods of the aspect graph (cf. *Braun [1994]*). Within this model the objects are projected into the image plane which can be brittle. In most of the aerial image the pitch angle from nadir is small so that the extraction of the tripod-configuration are not very distinctive. In *Kulschewski [1999]* oblique aerial images are used to overcome this deficit. Also the use of point-like information, e.g. with SIFT-features has shown that due to clutter and vegetation the desired objects in the image are the most homogeneous and stable regions, where in turn the least number of point descriptors are found. In this work a segmentation that builds a complete partition of the image and its geometry, topology and color information is used.

Figure 1.4 shows the hierarchy of the interpretation scheme using the Bayes net. Given the image, the extracted image features are represented in a graph, which is the feature adjacency graph. The image feature extraction delivers a complete partition of the image with adjacent regions. For each of the regions a random variable will be instantiated that represents an object-part node in the Bayes net. These object-part nodes are modeled as dependent nodes of the level above which are the object nodes. There are one or more object-parts dependent of an object node, but each object-part is

associated only to one object. The objects themselves are dependent on their neighbor-objects building a graph. Every object node is dependent on the random variable that instantiates the scene node, that acts as prior information about the type of the scene. The nodes of the features (not shown here) represent the observations that are carried out in the feature adjacency graph. They are modeled as continuous or discrete variables but their observations are numbers that do not have to be interpreted directly. The object nodes and the scene node have labels that are taken out of the ground truth database (visualized here as character). These labels are given by the human operator and his acquisition rule respectively. In contrast, the object-part nodes are not given. These manually created labels are stored in a ground truth database. It stores object labels, e.g. saddle-roof building, but not its decomposition into parts, e.g. roof window or chimney. The labels of these nodes are found automatically during the learning procedure. These found labels are shown as different colors. In Chapter 6 an interpretation of these labels is shown.

1.3 Prerequisits and assumptions

A good approach for an image interpretation model has to meet several reuirements. We used the following requirements to develop the model. The model should be:

semantic: The system has to represent spatial relations of objects. These are containing information like "streets are in the neighborhood of buildings". This context enables us to distinguish several classes of objects with similar image features. Therefore the model should be semantic.

region based: The underlying object model shows, that many objects of the human environment are planar. Highly structured objects like in *Bastian Leibe and Schiele [2004]*, *Weber et al. [2000a]* or *Helmer and D. Lowe [2004]* are not handled. The structured regions are recaptured again when they are smoothed in the higher levels of the scale space (c.f. *Lindeberg [1996]*). Our model should therefore be region based.

teachable: To avoid errors and to create a model that is adequate to the strength of the data, the model should derive the model from a typical subset of the data. It is necessary for the model to be teachable by a supervising user.

hierarchic: The system has to build the learned interpretation knowledge from bottom up. Many image features give evidence for the object part. Multiple object parts

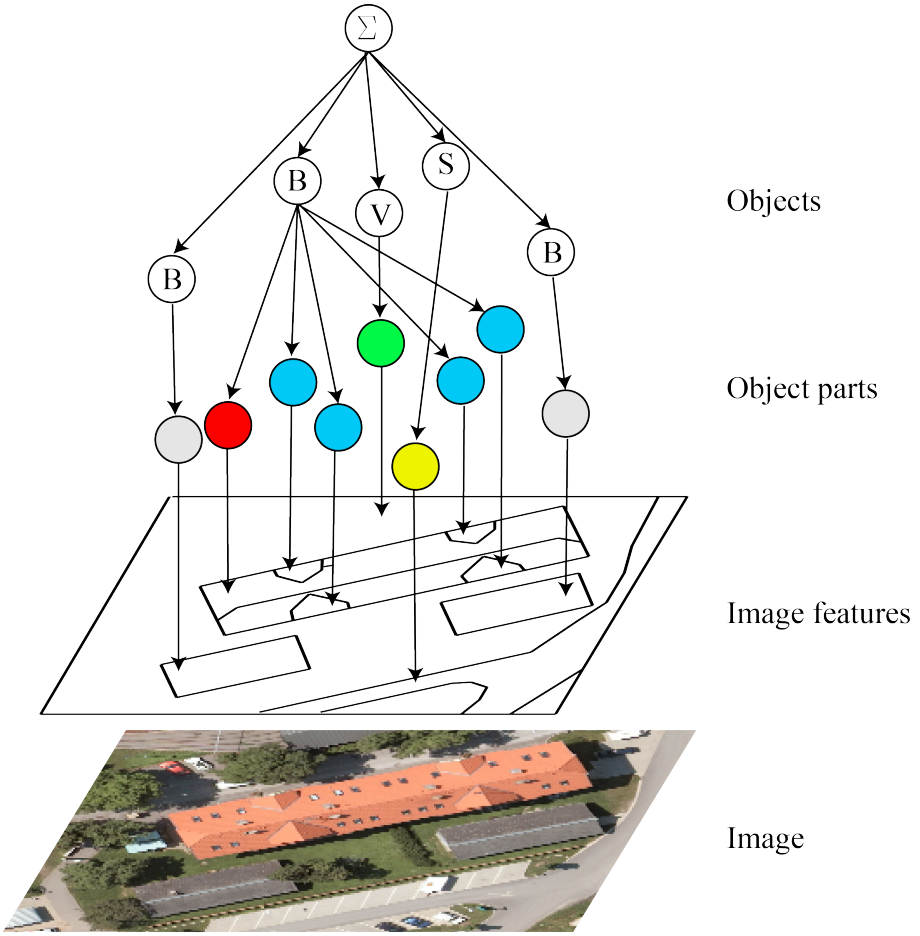


Figure 1.4: Layout of the interpretation scheme with the Bayes net. The labels of the object-nodes are defined in the ground truth database (S=street, B=building, V=vegetation) during the learning step. The labels of the object-parts instead are found automatically as visual words. They have no human readable labels and are here coded with colors instead.

build one object and several objects yield a scene. Vice versa it is possible to influence the detection by introducing knowledge at a higher level e.g. the scene information. Prior knowledge can be used to act top-down. The model should represent its knowledge in a hierarchic way.

dynamic: The number and location of objects can vary. We consider to model constant image partitions e.g. rectangular regions that are classified separately as not useful. The objects in an image are of varying size and orientation. They follow different spatial relations. Different to terrestrial images, in aerial images a certain arrangement of the objects in the image plane can not be assumed. To handle these variations in number and location, the model should be dynamic.

flexible: The system should be able to handle different kinds of image domains, e.g. aerial and terrestrial images.

robust: The model has to deal with suboptimal results as input from the underlying feature extraction process. Although there are many good feature extraction algorithms, each of them performs suboptimal if it is working out of their specification. Since the parameters can not be tuned for every image, the interpretation has to work with this uncertainty of the feature extraction.

Every single point itself can be met in a quite simple way. To fulfill the combination of these requirements, it needs a felxible and powerful model.

1.3.1 The special content of aerial images

Aerial images are quite special due to their orientation, size and resolution. For the image interpretation it is necessary to detect an unknown number of objects in a large image. The desired objects have some properties that make the detection model more complicated. First there is no prior information about the position or orientation of the objects. In terrestrial images objects are mostly under the sky and above a ground plane, orientation is induced by gravity.

Second, the desired class “building” has a big intra class variety. Even for humans the class “building” is not easy to define. Especially older European cities have very complicated roof structures. As roofs normally are not seen from the street, these are often no tidy structures.

Finally due to image noise and occlusions the feature extraction gives only suboptimal results. Even if there exist a lot of feature extraction algorithms, every one has its own special domain where it works best.

Without tuning the parameters in a classification process the results will be only suboptimal for an average real photo. The solution on which many detection and categorization approaches in recent years have built on, were rotation and scale invariant operators. These operators do not provide good results in our case because the image contains typically much vegetation. In the specific image scale of most aerial imagery they find extremely well points in the highly noisy and self similar vegetation but not on roofs. A challange is the clarity of the used ontology for the ground truth.

1.3.2 Image and object model

The objects we want to detect in the scene are man-made objects. We follow the work of *Braun et al. [1994]* who propose a model structure shown in fig. 1.5. Here, the correspondence of two-dimensional and three-dimensional models and their instances is constructed. On the 3D side a scene contains objects that decompose into object-parts, feature-graphs, features and eventually into voxels. The corresponding model to the objects are aspects that are position dependent views of the object. These decompose into aspect-parts, e.g. tripods of boundary lines *[André Fischer et al. 1998]*.

Unlike other researchers, e.g. *Kulschewski [1999]* and *A. Fischer et al. [1999]*, we favor a fully two-dimensional approach, which leads to a detection and a boundary reconstruction according to the underlying segmentation. The transition to the 3D-description, the right side of fig.1.5, is moved to a postponed reconstruction step. In contrast to the proposed model structure of *Braun et al. [1994]* stay in on the left side of fig. 1.5. Instead of the aspects we use objects and object-parts for the aspect-parts. The object model does not know anything about depth, so occlusion and perspective are pushed into the uncertainty of the detection. We use a phenomenological description of the scene. This is possible because we assume that deformations due to the projection into the image plane are small. On the other hand the learning step concentrates on features and regions that are stable. This leads to the effect that the extracted image features are invariant to the viewpoint.

The resulting image model conforms to the framework of *Förstner [1994]*. We assume that the objects are geometrically and physically bounded. In the digital image we have to deal with image noise as well as image distortions and deformations due to non-ideal cameras. This leads to an image description with regions, lines and points as image features as well as the mutual relations among those. In contrast to *[Fuchs 1998]* we use a image feature extraction that represents a complete partition of the image, i.e. every pixel is covered and there is no overlapping of image features and no background pixel

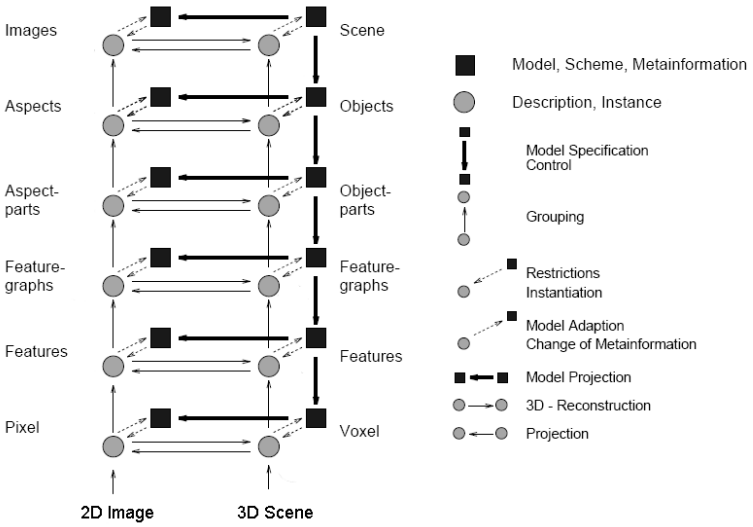


Figure 1.5: The structure of model hierarchies according to *Braun et al. [1994]*. Instead of modeling the 3D Scene, we use only the 2D model. The scheme acts phenomenologically and the model itself is learned from sample data.

between them.

1.3.3 Choice of image features

There is a big variety of different image features that can be extracted from images and can be used for detection. Every one of these has its own special domain where it is made for. If the feature extraction algorithm is used with different boundary conditions like different illumination, image noise or perspective, their result can be suboptimal. Without tuning the parameters, the results will be suboptimal for a randomly chosen image.

The image interpretation, for which the features are used, should work on many image domains. Therefore tuning the feature extraction is suboptimal in this case. Instead we use a group of different feature extraction algorithms. Their results are rated in the supervised learning step. By this rating we are able to distinguish between the cases and features that contribute to a successful interpretation and the ones that produce irrelevant features. This rating is performed in a statistic way.

Due to the image model we use, the backbone is a region-based feature extraction. These regions are used to extract a neighborhood graph of the tessellation. Every region is described by multiple descriptors. Line features and their attributes are used to describe

the boundaries and multiple attributes of the regions are used.

The result of the feature extraction step is an uninterpreted image description in form of a hierarchical region graph.

1.3.4 Choice of graphical models

After extracting the hierarchic feature graph, we need a method that is able to infer the inherent content of the feature graph to produce an interpretation of the image content. For this method we have chosen a graphical model in the form of a Bayes net. The choice of a Bayes net enables to meet the requirements mentioned 1.3 and has advantages over other approaches:

With Bayes nets we are able to infer, i.e. we can conclude from many weak occurrences to strong decisions. That allows us to detect objects in the image. Bayes nets represent not only a static design. They can be modified in structure und densities. It is possible to learn both, the underlying probability distributions and the conditional dependencies of these distributions as well as combinations of it. We will use both types of learning although the learning of the structure of the net is only applied on the region and clique node level.

We restrict the Bayes net to have a tree structure. This allows for the fastest and easiest way for inference algorithms of Bayes nets. It also reflects the natural object structure for image interpretation that forms a pyramid since the image features form a 2-d tessellation: The lower levels instantiate the image features. The image observations are introduced as observed nodes. Above these are the levels of the region-, clique- and object nodes. The top node represents the type of the image scene itself.

It is possible to follow the interpretation process in the Bayes net in every step. The dependency of a parent and child node for inference in Bayes statistics has always an interpretation of cause and implication *[Pearl 2000]*. In Bayes nets it is easily possible to do a dynamic augmentation of nodes during the interpretation process. This helps to manage the inference with a variety of net configurations.

Although we have chosen the Bayes net for this work, there are also other promising approaches that could be used to learn this kind of detection.

One of these approaches is boosting, where a strong classifier is learned by many weak classifiers. This algorithm has much in common with the naive Bayes classifier. The boosting framework is used in *[Elkan 1997]* to learn classifiers for detection. Also in *[A. B. Torralba 2003]* a boosting approach is successfully used. Neuronal networks are very similar to Bayes nets. They are used in a lot of works, an overview can be

found in *[Ripley 2007]*. Also the discriminative algorithms are used to represent learn able algorithms, e.g. K-means-clustering and support vector machines are used for image interpretation.

1.4 Contribution

We choose a statistical approach to cope with the following issues that we have found during the work with real world image data and feature extraction algorithms:

- The optimal feature extraction algorithms are not given and/or the parameters for the image processing step are not given in a way that would lead to optimal and error free feature extraction in sense of a human scene perception. These two error sources lead to a set of extracted features that can contain
  - too many features, e.g. at places where they should not be extracted,
  - too few features, e.g. features existing in the image could not be extracted,
  - features of low confidence, e.g. features are extracted not in the exact place or with a wrong occurrence.
- The basic set of observations which is used to learn from has deficiencies, e.g. wrong types of classes, missing features and partly occlusions.
- There is a set of different kinds of features that can be extracted from an image. These must be examined weather they are able to help the detection or not.
- Context provides information. Often the interpretation of existent features are ambiguous. Only the context information out of the neighborhood can lead to the right interpretation. These information can be of the type "buildings are in the neighborhood of streets".

These assumptions are explained in more detail in the subsequent chapters.

The approach uses a Bayes net that is learned from a database. For the learning process relatively few samples are used.

We present a pure 2 dimensional approach. Although — especially for buildings — the 3D reconstruction is of interest, the localization and outline reconstruction in 2D is an important step toward the reconstruction of the 3D structure. In a second step the 3D reconstruction algorithms can work on the identified and isolated areas.

1.5 Outline of the thesis

Chapter two presents related work in the areas of object detection, scene categorization, building detection and reconstruction. The chapter three contains the fundamentals about graphical models in general and learning Bayes nets in particular. This is followed by the chapter about the feature extraction algorithms that are used for the detection. In chapter five we develop the model for the detection. This model is tested in several experiments, presented in chapter six. A conclusion is drawn in the last chapter.

Chapter 2

Related methods

Object detection and scene interpretation is one of the key tasks in Computer Vision and Photogrammetry. It therefore has left a significant imprint in the vision publications. There are many different approaches and different combinations of algorithms of which some are introduced in short.

The related work we present here has many aspects under which it can be compared to each other. First of all there is the goal of the approach. As stated before, this can be an image categorization, an object detection of one or more objects with or without localization or even a reconstruction of the scene. Other aspects are the type of representation and tightly bundled the type of the used algorithms. Also the used image features, the type of knowledge representation and the original input data will be examined in short in this chapter.

2.1 The strategy

One of the basic ideas with object recognition has been formulated by *Fischler and Elschlager [1973]*. Their concern is the representation with part based models. The main idea is that relations between distinct regions contribute to the recognition (fig. 2.1). Therefore image features as 'meaningful parts' of the object are identified in the image. Relations between these parts allows to infer the main object in the image. The representation of the meaningful parts can vary as well as the representation of the relations. It is important that the recognition always relies on image features and geometric (projective distorted) relations between them. Every image feature gives some evidence that there could be an object in the image, but only the community of features let us conclude the presence. Especially for the case of multiple objects in the scene this

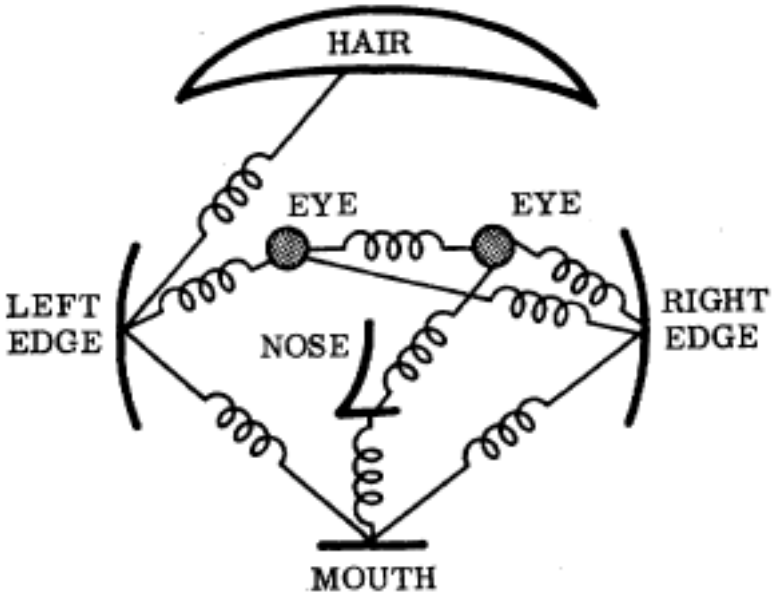


Figure 2.1: The model for recognition after *Fischler and Elschlager [1973]* identifies meaningful parts in the image. The model of the object contains relations of these meaningful parts to each other, which identifies the object in the image.

becomes relevant.

Another basic work is *Biederman [1987]*. He provides a model how the human visual system could detect objects. He introduces the concept of *GEONS*. These are small 3D objects of simple geometry. They are used as a vocabulary by which almost every man-made object can be composed by CSG-operations (fig. 2.2). The object parts and the composition operations are easier to model and parameterize than the complex model for the whole object.

Sharing common atomic parts enables the modeling of the high number of 10.000s of classes that humans are able to distinct. This divide and conquer strategy has influenced many others, e.g. the part-based approaches rely on this. The important difference to *Fischler and Elschlager [1973]* is that here the features (i.e. geons) are generic features that do not have a specific meaning. Their contribution is context sensitive. The cylinder can be part of the neck of an mammal as well as an vase or a trunk of a tree. In a constructive sense this helps to quickly build objects like in a CAD program out of a vocabulary of generic parts. For the recognition task this is an extra dimension of freedom which has to be solved. These basic ideas can be found in many approaches of recognition. It is the divide and conquer strategy that makes it successful.

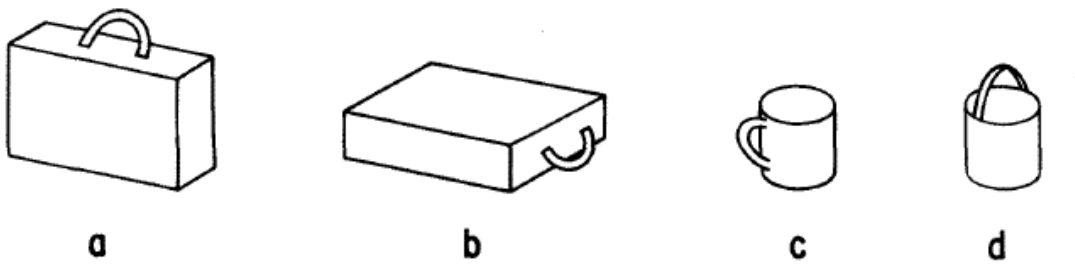


Figure 2.2: The GEONS are used as a CAD like vocabulary to construct real life objects. *Biederman [1987]* The interpretation of the same object parts depends on the arrangement, i.e. the context.

In addition to this, *Fischler and Elschlager [1973]* use more specific image parts which have also geometric constraints to their neighbors to detect complex object in a scene. This idea is the basis for the visual words approach that can be seen often. Here, image patches are used as a "'vocabulary"'. Like words in a language can form sentences to describe objects, here these visual words are used as visual terms or a "'bag of words"'. The visual words can be any image feature like shapes *[Burl et al. 1998]*, image patches *[Weber et al. 2000a,b]* or SIFT features *[S. Agarwal and Roth 2002]*. The disadvantage is that the parts can be distributed over the image, so we can not distinguish multiple objects or objects that are represented by similar features. This leads to an image categorization for the most prominent object in the image. To detect single objects in a scene, the context in recognition is important. *Rabinowich et al. [2007]* provide an additional cooccurence information in a conditional random field for their visual words approach. This helps to distinguish similar image parts from different objects. Other researchers introduce the spatial arrangement of the features *[Carneiro and David Lowe 2006; Crandall et al. 2005; Fergus et al. 2005; B. Leibe et al. 2004]*. Here the relative position among the features or against a common center point are modeled. This helps to get multiple features as well as their orientation and position in the image. These properties are important to our image interpretation with many similar objects in aerial images.

The idea of Biederman follows the work of *A. Torralba et al. [2004]* who emphazies the importance of sharing the same features for the efficient representation for recognition of objects. This plays a big role in learning the representation in an efficient way. *L. Fei-Fei et al. [2003, 2004]* and *L.-J. Li and Li Fei-Fei [2007]* show that it is possible to teach generative models from few samples. This is important to get to an incremental

learning.

An interesting approach to shift the scene interpretation to a desciptive level is presented in *Farhadi et al. [2009]*. The researchers use image features to learn textual desciptive attributes wich describe the scene. Using this extra layer, the image interpretation can operate on a more human compatible descriptive way. *[Russakovsky et al. 2013]* presents an analysis of the status and the next steps regarding object detection and localization in image databases.

2.2 Image features

The image as a set of pixels coded as gray values is difficult to address for part based interpretation. This is why most approaches use some kind of feature extraction to work on a more abstract symbol level than the pixel level itself.

The approaches that use Markov random fields are able to work on the image pixels themself *[Klonowski and Koch 1997; Meidow 2000]*. With direct pixel observation in a classical Markov random field or HMM the complexity of the scene is often very limited or has to build a hierarchy to model more abstract objects *[S. Kumar and Hebert 2003]*.

The basic image feature that is widely used is the edge or line. This can be a contour line of the object, e.g. generated by a blur descriptor *Berg et al. [2005]* or level sets *Rosenhahn et al. [2006]*. The contour line can be matched to known objects by using the curvature, length or color information *Belongie et al. [2002]; Mikolajczyk and Cordelia Schmid [2003]; Rodrigues and Albuquerque Araújo [2002]*. To handle view point and intra class variations, several researchers formulate a generalization of the shape, e.g. over a contour network defined by intersection of contour lines *Ferrari et al. [2006]* or graph matching like in *Dickinson et al. [2005]* or *Tu et al. [1999]*. The use contour line works well for recognition as long as we have only few distinct objects with no occlusion in the scene. For our task of interpretation of elements this is not applicable.

In contrast to the closed contour lines, straight line elements are often used to find geometrical objects in scenes. This approach can be seen in many approaches for building reconstruction like *C. Lin and R. Nevatia [1995, 1996]; Chungan Lin and Ramakant Nevatia [1998]* or *Noronha and Ramakant Nevatia [1997, 2001]*. They use the extraction of long edges as features. The edges are grouped and used to build hypotheses about geometric structures like rectangles, parallelograms and trapezes. These are matched with geometrical models to find buildings and reconstruct them as 3D model (*Collins et al. [1998]; Paparoditis et al. [1998]* and *Jaynes et al. [1997]*).

The approach with straight lines is not suitable for objects that are of irregular

shape. In this case correlation of image patches *[Amores et al. 2005]*, color histograms *[A. Agarwal and Triggs 2006; Chang and Krumm 1999]* or texture *[Y. Li et al. 2005]* and entropy *[S. Kumar and Hebert 2003]* can be used.

Important image features are point features to identify salient points in the image. The points themself are mostly found by Harris or Foerstner point detectors, but also based on stable regions *[Matas et al. 2002]* or entropy *[Kadir and Brady 2001]*. Descriptors like the one of *D. G. Lowe [2004]* bring the advatage that they are invariant to some distortions like scale and rotation. This helps to cut down the search space for the matching. The big number of approaches shows the importance for object recognition and detection, e.g. *Helmer and D. Lowe [2004]; Loy and Eklundh [2006]*.

Descriptive image patches and regions can be found by the segmentation of the image. Segmentation approaches are published for example in *[Vogel 2004]* and *[Felzenszwalb and Huttenlocher 2005]*. *[L.-j. Li et al. 2010]* introduce a meta feature layer. They use a filter bank like they are used in texture detection. Here, these filters are trained to objects and can be of various kinds, here represented by an trained SVM object detection and a texture fileter. The answer from the filter bank over a scale space of the image is then fed to a classifier to carry out the scene interpretation itself.

2.3 Models and algorithms

Model and algorithm are very closely interlinked that in most models they cannot be separated.

For the recognition and detection of man made objects, especially objects that have long straight lines like buildings or indoor legoland scenes, geometrical reasoning is used in many publications. Therefore, egdes and lines are grouped together to geometrical structures like rectangles, trapezoids etc. These are matches against the geometrical model to find the best hypotheses. Especially for the use of building detection, 3D knowledge is used in the model that is validated against stereo images to infer the detection.

Graph matching is used in approaches that have organized the image features in a graph structure. *Tu et al. [1999]* use for example a graph matching to recognize objects by identifying aspect parts with image segments. The problem of invariance against projective deformations is addressed in *Dickinson et al. [2005]* and *Bangham et al. [1999]*. The merging of regions by following the region over the scale space creates a graph structure that is matched against a learned graph structure.

The clustering of high dimensional feature vectors with k-means *[Philbin et al. 2007]* or nearest neighbour *[Aly et al. 2011]* algorithms can be very fast due to effecting indexing

algorithms. The key is to find the appropriate and lowest possible dimensionality of the feature vectors *[Jegou et al. 2012]*.

Other discriminative models are found in the literature. Especially those models that are easy to train like boosting *[Zhang et al. 2005]* or support vector machines *[Dorkó and C. Schmid 2003]* or probabilistic maximization approaches *[Burl et al. 1998]*.

Many approaches use graphical models for recognition. The Markov random fields or conditional random fields often have been used in low level vision, e.g. for image segmentation. But it can also be applied for pixel labeling *[Meidow 2000]* and image segmentation *[Boykov and Jolly 2001; Korc 2012; Rother et al. 2004]*. There are some ideas published that try to overcome the limitation of the Markov fields to be restricted to the regular 2D grid of the image plane. If the Markov field is not defined in the level of image pixels but one level higher in abstraction, it can be used to represent image patches and regions, e.g. *Drauschke [2011]; A. B. Torralba [2003]* where the conditional random field is learned on a region adjacency graph structure to obeserve the color and texture.

The nature of the MRF that the nodes are horizontally dependent on their neighbor nodes is a big advantage because it helps to identify connected features in the image that belong to one (rigid) object and can be seen as some spatial constraint. With this it is possible to detect multiple instances of objects in the image.

Bayes nets in general do not have that feature, but it can be augmented in the model. Instead they are more flexible to model the scene content, i.e. the hierarchy of objects and object-parts and their spatial context. Examples for these spatial terms in a Bayesian net are found in *L. Fei-Fei et al. [2004]; Niebles and Li Fei-Fei [2007]; Weber et al. [2000b]* or *Cao and Li Fei-Fei [2007]*. Also factor graphs as a generalization of graphical models are used for scene interpretation, e.g. in *Yang [2011]*.

2.4 Prior knowledge

The result of a recognition and image interpretation can be enhanced by prior knowledge about the object or scene which is not hard coded in the model. This may be the position, posture or number of objects. There are multiple ways how to integrate such a prior knowledge. In *K. Murphy, A. Torralba, Eaton, et al. [2005]; Oliva and A. Torralba [2001]* the authors introduce the GIST, a low dimensional abstraction of the image which is calculated by a pricipal component analysis over the pixel domain. This is used in *K. Murphy, A. Torralba, and Freeman [2003]; A. B. Torralba et al. [2003]* to initialize the scene interpretation with prior knowledge and steer the interpretation in this way. In

Oliva, A. B. Torralba, et al. [2003] this method is used to introduce a top down control of visual attention. The *GIST* provides in this context a saliency map where to look first in the image. Also in *M. Kumar et al. [2005]* there is a very basic segmentation introduced as prior knowledge into an HMM for labeling the image and infer the segmentation.

L.-J. Li and Li Fei-Fei [2007] show that the scene type interacts with the object detection but can also be inferred from this. Other approaches like *Vogel and Schiele [2004]* or *Schröder-Brzosniowsky [1999]* use a very small image scale to get a scene categorization. This can be used to influence the object detection or interpretation.

Additional sensor data is also used to improve the results of scene interpretation tasks. In a classical way, the height information given by radar or a general digital elevation model *Dissard et al. [1997]; Rottensteiner [2003]* can be introduced to improve the inference. In the recent years th availability of point clouds coming from dense image matching or LIDAR are taken to detect also the 3D structure of scenes *Zia et al. [2013]* and buildings *Nguatem et al. [2013]*.

Chapter 3

Theoretical basis

3.1 Notation

Throughout the text we will use the following notation:

x	random variable or its corresponding node in the Bayes net
$\boldsymbol{x}$	set of random variables or nodes
$P(x)$	probability of x
$p(x)$	probability density of x
$E_{p(\cdot)}(\underline{x})$	Expectation of x regarding $p(\cdot)$
$\mathrm{pa}(x_i)$	the set of parents nodes of x_i
$\mathrm{ch}(x_i)$	the set of children nodes of x_i
$\mathcal{O}$	set of indices
$\mathcal{M}$	the model, i.e. the graph structure
$\mathcal{D}$	set of data
$\mathcal{T}$	set of test data
$\boldsymbol{\Theta}$	a vector of probability densities
$x \perp y$	two independent probabilty densities

3.2 Probability theory

The data that is produced by the feature extraction step results in a set of data of the general form $\mathcal{D} = \{x_1, \ldots, x_n\}$, where the x_i are vectors of observations. We use statistical methods to derive conclusions on the nature of the underlying process that produced these observations as well as the expected values for some future events. The probability model for our interpretation is a multidimensional probability distribution

Name	Probability Density	cpd
Beta	$p(x) = \text{Beta}(x \mid \alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma\beta} x^{\alpha-1} (1-x)^{\beta-1}$	
Binomial	$p(x) = \text{Bin}(k \mid p, n) = \binom{n}{k} p^k (1-p)^{n-k}$	Beta
Multinomial	$p(x) = \text{Multi}(\underline{x} \mid \boldsymbol{\mu}) = \prod_{k=1}^{K} \mu_k^{x_k}$	Dirichlet
Poisson	$p(x) = \text{Poiss}(x \mid \lambda) = e^{-\lambda} \frac{\lambda^x}{x!}$	Gamma
Gaussian	$p(x) = \mathcal{N}(x \mid \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp -\frac{1}{2} \left(\frac{x-\mu}{\sigma}\right)^2$	Gaussian

Figure 3.1: Probability distributions with their parameters and the conjugate prior distributions (cpd).

that we first estimate out of the data and later use for inference of evidence. All derived statistical conclusions are conditional to this assumed probability model.

We use the Bayesian statistics in a sense that we use conventional probability theory in the context of the axiomatic introduction by Kolmogorov and the interpretation of Bayes rule for inferencing.

The classical definition of the probability P for the discrete random variable X_i to yield a specific result x_i is denoted as

$$P(x_i) = P(X_i = x_i)$$

.

3.3 Bayes nets

A graphical model is a probabilistic model that factorizes according to an underlying graph. As such graphical models are a graphical representation of a joint distribution over a large number of random variables by a product of local functions that each depends on a small number of variables.

Graphical models describe conditional independencies of the random variables in a graphical scheme. This graphical scheme can be used to visualize the structure and reduce the computational complexity of inference in the model. According to the graph and the formulation of the local factors, we can distinguish between Bayes nets that are based on a directed graph and the Markov random fields with an undirected graph. Both types can be formulated as a factor graph *[Bishop 2006]*.

The factorization of the joint distribution of the example net shown in Fig. 3.2 can

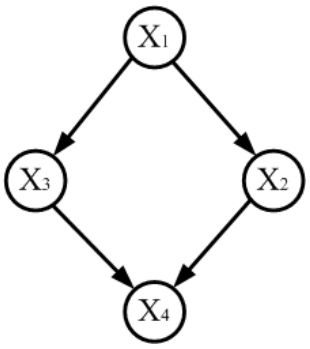


Figure 3.2: A small Bayes net modeling four nodes. The nodes x_2 and x_3 are independent from each other, as well as x_1 and x_4.

be written as

$$P(x_1, x_2, x_3, x_4) = P(x_4|x_3, x_2)P(x_2|x_1)P(x_3|x_1)P(x_1).$$

The graph structure models the independence of the variables. The Bayes net consists of the variables x graph structure $\mathcal{M}$ and the parameters for steering the probability distribution Θ.

The preferred graph structure for Bayes nets is a directed acyclic graph, for which an exact inference solution is possible. In this case the joint distribution factorizes as

$$P(x_1, \ldots, x_n) = \prod_i P(x_i|\,\mathrm{pa}(x_i))$$

where $\mathrm{pa}(X_i)$ denotes the parent nodes of the node x_i in the graph.

Often Bayes nets consist of repeating structures. Instead of writing the Bayes net with all nodes like in Fig. 3.4, the plate writing scheme can create a better overview. Therefore we write multiple dependencies of similar nodes as a plate and indicate the number of nodes on that plate (see fig 3.5).

The variables in a graphical model can be of various probability densities (e.g. fig. 3.1). If the parents of a node have other probability densities than the conjugates of the node, there exists no closed form for the posterior, so inference can only be approximated. This can be numerically difficult.

Noninformative priors

It is an often observed phenomenon that if there is a lot of strong data the model seems to emerge from the data without help, independent of any prior information. If the data is weak or there are too few samples, the model becomes highly dependent on the a priori knowledge in the Bayes net. This behavior will occur if the samples contain not enough information to reproduce the characteristics of the data. For learning graphical models, it can happen that the training data is weak or that during an automated model search the model does not fit the data. In this case, the knowledge that is incorporated in the a priori distribution prevents an improvement of the results.

To avoid a burning-in of the probability density function by a prior we have to introduce a so called *non-informative* prior. This enables us to model the knowledge that there is no knowledge so far. *Bishop [2006]* gives an introduction of non informative priors for the Gaussian family. In most cases it is a good approach to model the prior like a uniform distribution, e.g. to initialize the Gaussian with a very high variance.

3.4 Inference

In a recognition task there are some nodes of the Bayes net that are observed i.e. they are bound to fixed values that are introduced into the Bayes net. In the graphical formulation we note these nodes as shaded (see fig 3.3). Given these observations it is of interest how the probability of the unobserved nodes behave. Therefore we wish to compute the posterior distributions of the other variables in the net.

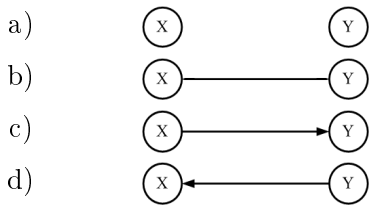


Figure 3.3: Two random variables: a) $x \perp y$, b) joint probability $P(x, y)$ c) conditional probability $P(y|x)$ and d) $P(x|y)$

There are many algorithms proposed for inference that can be divided in exact inference and approximate inference approaches.

In general Bayes net graphs which can have loops, the processing can be complex. Here we can use the junction tree algorithm, loopy belief propagation *[Ihler et al. 2005]* or the generalized belief propagation *[Broadway et al. 2000]*.

For singly connected networks like trees there are algorithms like message passing and belief propagation *[Pearl 1988]*, arc reversal or subtree conversion *[Lauritzen 1996]*.

The main idea behind most of the inference algorithms is the scheme of local message passing. This says that every node sends a message to its parent nodes and children. The message to the parents consists of the probability of every state of the parents given the evidence of the observed children nodes ch(x) and the node x itself. The message to the children is then instead the probability of the node x given the evidence observed in the parents pa(x). In graphs that do not have loops, e.g. tree structures, this behavior divides the graph into two half, the parents and the children of node x. The marginal distribution of a node is then proportional to the product of the messages of the parents, the message of the children and the probability of the node itself.

$$P(X|\text{evidence}) \propto \begin{matrix}\left[\sum_i P(X|\,\text{pa}_i(X))\prod_j P(\text{pa}(X_j)|\,\text{pa}((\text{pa}(X_i)))\right] \\ \prod_k P(\text{ch}_k(X), \text{ch}(\text{ch}(X)), X)\end{matrix} \tag{3.1}$$

where the integration has to be done over the complete parameter space of node X. Often the path of influence of the message passing can be limited due to d-separation *[Buntine 1994]*, but in general the problem of exact inference is NP-complete *[Heckerman et al. 1995]*.

To reduce the computational load the inference can be approximated like Monte Carlo Markov Chain methods *[Pradhan and Dagum 1996]* or variational Bayes inference *[Jordan et al. 1999; Winn et al. 2005]*. An approximation that combines the simplicity of message passing with the variational Bayes inference is the variational message passing algorithm *[Winn et al. 2005]*. The inference and learning of graphical models is the topic of many approaches. *Lauritzen [1996]*, *Zoubin Ghahramani [1997]*, *Kevin P. Murphy [2001]* as well as *Almuallim and Dietterich [1991]* solve the problem to receive many not relevant observations that disturb the learning by using PAC-Algorithms *Valiant [2013]*. This is done by evaluating the bias and introduces this as a penalty term into the learning algorithm to avoid over fitting.

3.4.1 Plate writing

In Bayes nets there often exist repeating structures like depicted in fig. 3.4. To shorten this, another writing is introduced: the plate writing. With this, the net in fig. 3.5 is equivalent to the models before. The rounded rectangle with a number or variable in the lower corner represents the repetition of the contained structure of the net.

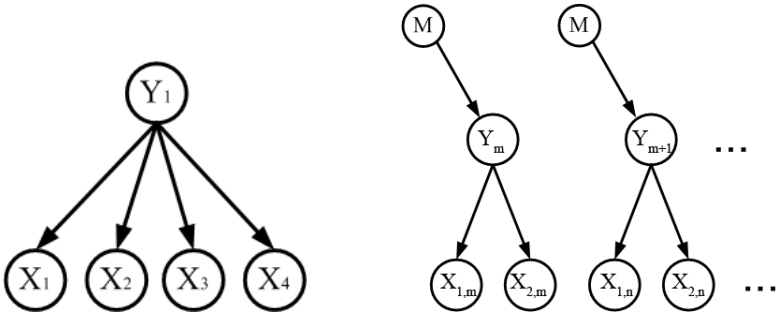


Figure 3.4: Repeated structures in Bayes nets

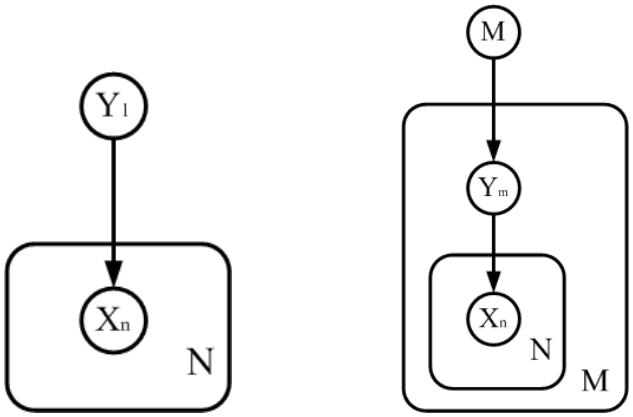


Figure 3.5: The plate writing. the nets are equivalent to the ones in Fig.3.4.

3.5 Learning

A graphical model contains the following types of information that must be provided to allow for direct inference or drawing samples: The graph structure, the types of the probability densities and their parameters. It is cumbersome or even impossible to manually specify the structure of the probabilities in real world situations. Therefore it is prefereable to learn these.

When we think in the Bayesian way, parameters can be modeled like new random variables. In this case, learning the parameters can be treated like the inference of hidden variables. In the non Bayesian thinking we carry out a point estimation of the parameters using a maximum likelihood or a maximum a posteriori search. Thus we can use the same algorithms of exact or approximate inference to learn the parameters in the fully observed case.

In the easiest case we have given the structure of the Bayes net and we can observe all random variables. In this case learning the Bayes net reduces to counting the states (for discrete variables) or integrating over the random vairables.

If it is not possible to observe all of the nodes during the learning step or the graph structure is unknown, the learning procedure becomes more complex because we have to integrate over unknown nodes. There are the following cases:

3.5.1 Learning parameters with unobserved data

In the case that the observations are not contained in the dataset or nodes are not observable at all like mixing variables we model these as hidden variables in the Bayes net. In the case of hidden variables, the decomposition like in 3.3 fails. Given an observed node Y and a hidden node X we get

$$\mathcal{L}(\theta) = \log P(d|\theta) = \log \sum_X P(Y, X|\theta) \tag{3.2}$$

for discrete variables.

We can solve this by using an EM-algorithm as follows: Introducing an arbitrary

distribution for the unobserved node $X \sim Q(X)$ we can derive a lower bound for $\mathcal{L}$:

$$\begin{aligned}
\log \sum_X P(X,Y|\theta) &= \log \sum_X Q(X) \frac{P(X,Y|\theta)}{Q(X)} && (3.3)\\
&\geq \sum_X Q(X) \log \frac{P(X,Y|\theta)}{Q(X)} && (3.4)\\
&= \sum_X Q(X) log\, P(X,Y|\theta) - \sum_X Q(X) \log\, Q(X) && (3.5)\\
&= \sum_X I(X,Y) - \sum_X H(X) && (3.6)\\
&= \mathcal{F}(Q,\theta) && (3.7)
\end{aligned}$$

where 3.4 follows from the Jensen-inequality. $I(X,Y|\theta)$ denotes the mutual entropy of the distributions P and Q, $H(X)$ the entropy of the distribution Q in X. $\mathcal{F}(Q,\theta)$ can be interpreted in physics as the free energy. We can now use the expectation-maximization algorithm (EM-algorithm) *[Dempster et al. 1976]* to maximize $\mathcal{F}$ with respect to Q and θ.

1) Expectation step: $Q_{k+1} \leftarrow \text{argmax}_Q\, \mathcal{F}(Q,\theta_k)$
For the maximization of the lower bound we get

$$Q(X) = P(X,Y|\theta_k).$$

2) Maximisation step: $\theta_{k+1} \leftarrow \text{argmax}_\theta\, \mathcal{F}(Q_{k+1},\theta)$

So we get

$$Q_{\theta_{k+1}} = \sum_h P(X|\mathcal{D}_h,\theta_k) log\, P(X,\mathcal{D}_h|\theta_{k+1}).$$

3.5.2 Learning structure with complete data

An insufficient modeled dependency can not be compensated by tuning of the parameters. Each additional arc leads to higher complexity and eventually to an over fitting; each missing dependency leads to deficits for modeling the data.

In the case we have given some nodes without knowing their interdependencies but with a fully observed dataset, we can continue a search over the space of models $\mathcal{M}$. With the given data we can integrate out the dependency of the parameters θ and get

$$P(\mathcal{M}|\mathcal{D}) = \frac{\int P(\mathcal{D}|\theta,\mathcal{M}) P(\theta|\mathcal{M}) d\theta P(\mathcal{M})}{P(\mathcal{D})} \quad (3.8)$$

to find the best fitting structure. If there is any prior knowledge about a preferred structure we can insert this by providing an informative prior about the set of models $P(\mathcal{M})$.

The search space over the model space is super exponential in the number of nodes so it becomes quickly very large. It is not easy to find an adequate search strategy for introducing arcs. There can be a local search strategy with a gradient ascent or a global search like simulated annealing.

Another assumption concerns the evaluation of the net structure. Here we have to define how to prefer one structure over the other. The maximum likelihood estimation can not be used to evaluate the best model $\mathcal{M}_{ML}$. Because the insertion of an additional dependency between two nodes leads to a higher scoring, the search would tend to insert as many arcs as possible, leading to a fully connected Bayes net. The result would be an over-fitted network.

To avoid this we can use a prior on the model $P(\mathcal{M})$ that is more probable for simpler models. Since such a prior is difficult to formulate, we can approximate this behavior. The solution here is to insert a term that punishes the increase of the network complexity. We use a scoring function that acts equivalently to a Minimum Description Length criterion. Therefore we can use the Bayesian Information Criterion (BIC) or the Akaike Information Criterion (AIC).

$$BIC : \log P(G|D) \approx \underbrace{\log P(D|G, \hat{\theta}_{ML})}_{\mathbf{A}} \underbrace{-\frac{d}{2} \log N}_{\mathbf{B}} \tag{3.9}$$

$$AIC : \log P(G|D) \approx \underbrace{\log P(D|G, \hat{\theta}_{ML})}_{\mathbf{A}} \underbrace{-d}_{\mathbf{B}} \tag{3.10}$$

Here we compare the increasing posteriori probability (A) with the punishing term for the net complexity (B), where d denotes the number of parameters and N the number of samples. The maximum likelihood estimation of the parameters is denoted as $\hat{\theta}$. Instead of the likelihood we estimate a maximal score by using the EM-algorithm.

The scoring functions have another useful property: They decompose according to the net structure. For the Bayesian Information Criterion we could formulate this as $BIC(\mathcal{G}|\mathcal{D}) = \sum_i BIC(X_i|Pa^{\mathcal{G}}_{X_i}, \mathcal{D})$. As a consequence, we do not need to recompute the score for large networks for every local modification on the graph structure. Instead we can cache the scores of the unchanged parts and reuse it during the search for an optimal structure *[Koller and Friedman 2009]*.

3.5.3 Learning structure with incomplete data

In the case that we have not completely observed data like in the sections before, computing the marginal likelihood is intractable because it is required to sum out the hidden variables as well as integrate out all the parameters θ

$$P(Y|G) = \sum_X \int_\theta P(X, Y|G, \theta) P(\theta|G). \tag{3.11}$$

The score here is not easy to compute and the term does not decompose easily into local terms like the statements above. The search for an optimal structure and its best parameter set has to be combined in one or another way.

There are two possible approaches for this problem. The first is to approximate the marginal likelihood and link this into a structure search algorithm, the second would be to use a special scoring function that decomposes.

Friedman et al. [1997] show how to model the search for the best structure as a Bayesian problem itself. They cut the graph in subgraphs and define a probability distribution over the existence of these parts. The structure is approximated then by a MCMC search through the model space, c.f. [P. Green 1995; Richardson and P. J. Green 1997].

Friedman [1997] develops a Structured-EM algorithm (SEM) that combines the structural search with the parameter search. It extends the EM algorithm to search alternately for the best fitting structure and then for the best fitting distribution parameters. First it computes a new set of parameters $\theta^{i'}$, then it evaluates all the graph structures G' of the Bayes net that are "'neighbored"' the actual graph G. The new neighbor graphs G' are assessed by evaluating the BIC score. The graph with the best score is set as the new actual graph structure and its parameters are updated in a M-step. Algorithm 1 shows the pseudo-code after [Friedman 1998].

Algorithm 1 A pseudo-code for the Structural-EM algorithm. Additional to the expectation and maximization step of the parameters, the graph structure is searched through and evaluated with a BIC-score. The structure with the best score is then taken as structure for the next iteration.

```
1:  procedure StructuralEM(G^0 as initial Graph, θ^0 as initial )
2:      i = 0;
3:      while not converged do
4:          estimate θ^i'                                   ▷ expectation step
5:          for all neighbor G_n^i of G^i do                ▷ modify the graph
6:              compute θ_n^i for new graph                 ▷ compute also here the Expectation
7:              compute s_n = BIC-score(G_n^i)              ▷ and compute the cost term
8:          end for
9:          G* := argmax_{s_n} G^n                          ▷ find the best match
10:         if BIC − score(G*) > BIC − score(G^i) then
11:             G^{i+1} := G*                               ▷ structural M step
12:             θ^{i+1} := θ(G*)                            ▷ parametric M step
13:         else
14:             converged := true
15:         end if
16:         i++;
17:     end while
18: end procedure
```

Chapter 4

Structure of the hierarchical region adjacency graph

One key issue for learning is the representation of the image features as input for the interpretation. It is not efficient to operate on the pixels themself when we want to detect objects in the image. This parameter space is high dimensional and due to the heavy over-parametrization mostly empty. Therefore we use feature extraction algorithms to code the content of the image more efficiently.

There is a big variety of different image features that are used in the computer vision and image processing community. Each of them has its own domain where it is used in an optimal way. Unfortunately most of the features have only little use if they are applied out of their original context and with different image conditions like image noise or illumination changes. Some feature extraction algorithms extract features that are invariant against several transformations like rotation or scaling, see *[D. G. Lowe 2004]*. These are very efficient in some context. There are no super-features that have equal interpretation under all conditions, so the detection step has to cope with suboptimal features.

In the work of *Fuchs [1998]* a consistent framework is presented that addresses point, line and region like features in a consistent map over an adjacency graph. We used this framework as prototype for the creation of a feature adjacency graph. The three features do not partition the image area completely but leave certain pixel unclassified. This is why we implemented an other partitioning scheme.

The image model that we use in the detection is presented in *Förstner [1994]*. It characterizes rigid objects that appear as more or less homogeneous regions that have a sharp contour line that defines its scope. To be able to describe this object model,

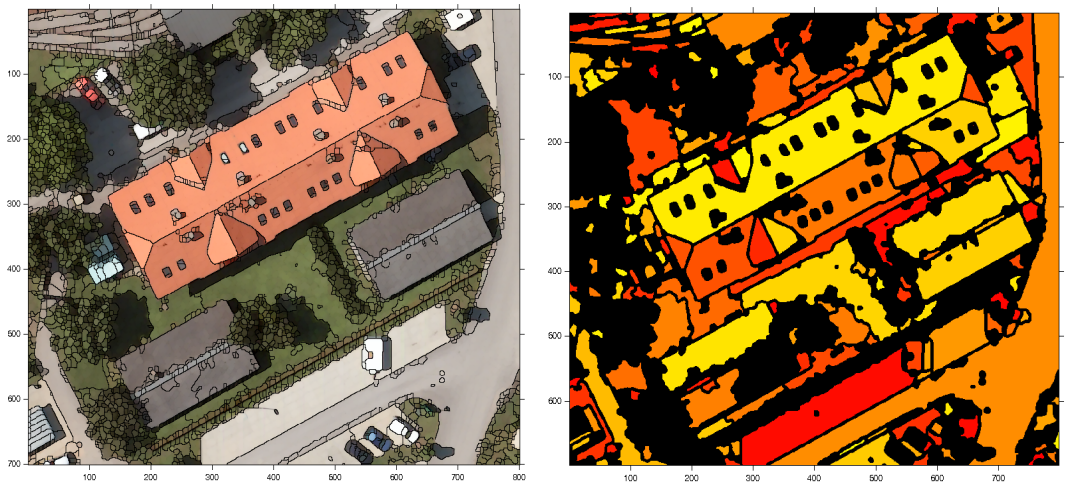


Figure 4.1: On the left the maximum stable regions in color space using the first part of the MSER algorithm is shown. The right image shows the extracted and separated regions. To show the result better, the regions are eroded by 1 px width and regions smaller than 20px are left out.

we define our adjacency graph as region based. All other features (i.e. lines, shape information etc.) are then related to the regions. We apply this image model to aerial images, but it can hold also for terrestrial images if the objects are man-made and occlusion is low.

4.1 Extraction of homogeneous regions

The algorithm used here is a derivative of the region detector MSER *Matas et al. [2002]* that extracts the maximal stable regions in the color space. The regions used here (see fig. 4.1) are maximum stable regions which are extracted by a watershed algorithm running on the squared gradient of the image. Here only one threshold is used to separate lines and regions thus the regions are separated by lines of one pixel width. The region detector gives a partition of the image, but tends to over-segment the scene. Especially in vegetation areas it has some problems. The extracted regions are listed in the regions set $\mathcal{R} = \{R_i\}$.

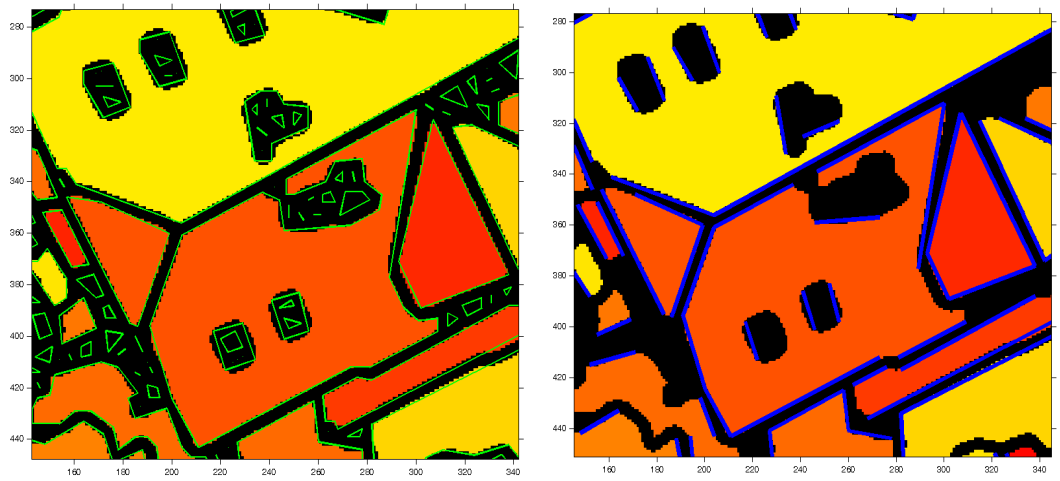


Figure 4.2: The image on the left shows the region boundaries of the extracted regions. Out of these edge-chains, the straight lines (here with a length > 25 pixel) are extracted (right image). These are stored as feature properties per region in the feature adjacency graph.

4.2 Extraction of lines and points

In this work, the boundary regions of the region extraction described above are not taken as edges. Even if the region detector has detected two regions, it can happen, that the difference is that small that there is no edge between the regions. Therefore we use an extra edge detection that follows the boundaries of the extracted regions.

The binary pixel results boundary operator are lined up to edges and are stratified by a Peucker algorithm *[Douglas and Peucker 1973]* to remove jitters and disturbing effects. The result is a net of lines, connected over junctions. This is fitted later in the feature adjacency graph (see 4.4) by attaching them to the corresponding regions. The set $\mathcal{L}_r = \{L_j\}$ contains all extracted lines for the Region r. The boundary lines that are between two regions are introduced separately for each region into the bayes net as independent observations.

4.3 Features over scale

The motivation for the use of scale spaces is the fact that most real-world objects exist only on a certain range of scale. The use of different levels of scale is well known in cartography. The making of maps makes heavy use of the knowledge of scale spaces.

There are different approaches to create scale spaces. The most common is the Gaus-

sian scale space where a Gaussian kernel is used to smooth the data iteratively *Lindeberg [1990]*. Other approaches use morphological filter sets, or, to define the problem on multi channel data, a minimum-maximum filter, cf. see *Mayer [2000]*, *Bangham et al. [1999]* or *Harvey et al. [1997]*. Additionally there are continuous problem approaches which are solved by e. g. using diffusion equations *Clarenz et al. [2004]*. We will use the Gaussian scale space for this work.

In this context we use the scale space to examine scale space events for those regions, i.e. the merging of regions. In contrast to other work on scale space events our interest is not only the change that the event creates. Here the emphasis is on the stability of the existence of regions between the events. The stability of manmade objects in aerial images has been investigated by *Drauschke et al. [2006]*. The result of this work is that manmade objects have a longer existence over the scales than natural features.

For creating the scale space we follow the approach of [Crowley et al. 2003]. Therefore a Gaussian scale space is introduced. The pyramid levels are defined by

$$g_c(x, y, \sigma) = g_c(x, y) * \frac{1}{2\pi\sigma} e^{-\frac{(x^2+y^2)}{2\sigma}} \tag{4.1}$$

where g_c denotes the color information of each pixel and $*$ denotes the convolution. The scale space is discretely sampled at ten levels $\lambda = 1, \ldots, 10$. To spread the levels equally, the Gaussian smoothing is done with $\sigma = 2^{\lambda-1}$.

The lifespan of a region that we observe is defined as the maximum number of scales under which the region does not vary. Thus the segmentation provides a complete partition of the image $\bigcup_l Region_{l,\sigma}$ where l is the number of the region, so there are no overlapping regions inside one scale. A region is labeled to be stable if there is no merging of regions. Figure 4.3 shows the situation for regions and their boundaries. The chosen scale space creates not exact merging: due to the Gaussian filtering and the subsequent region extraction the boundary lines can move slightly without a complete merge. To overcome this, we approximate the scheme of 4.3. The stable regions are extracted by comparing two regions in adjacent octaves. If the non overlapping part is smaller than a threshold, the region is marked as stable:

$$V_{\sigma_i,\sigma_j} \in [\sigma_i, \sigma_j] : |R(l, \sigma_i) \setminus R(j, \sigma_j)| < t_s \tag{4.2}$$

The threshold is set to 5% of the merged region size. We do extract only the first region in the scale dimension, that reaches the stability criterion, since we want to stay in the over segmentation concerning the objects. Figure 4.5 shows the life span of the first 160

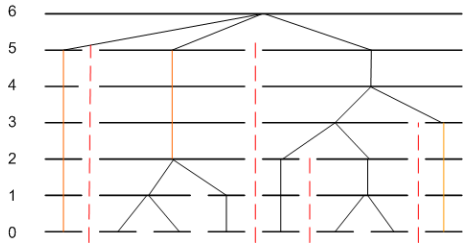


Figure 4.3: Schema of the scale space over 7 octaves. The periods between the merging events in scale space are periods of stability. These can be extracted for lines (dotted) and regions.

regions shown in figure 4.4.

4.4 Extraction of the feature adjacency graph

For the detection of objects it is necessary to know not only the shape of the examined object itself but also to know about its neighbors because a significant part of the information is provided by the context of an object. Therefore an adjacency graph is created that incorporates all the extracted features as nodes. The attributed graph $\boldsymbol{G}$ is defined as

$$\boldsymbol{G} = (\mathcal{V}, \mathcal{E}, f, g) \tag{4.3}$$

where the vertices $\mathcal{V}$ are the features and the edges $\mathcal{E}$ represent the neighborhood of the features. Both, the vertices $\mathcal{V}$ and edges $\mathcal{E}$ are attributed: $f(v_i)$ and $g(e_i)$. The vertices contain an attribute vector with observations that are made per image region. The edges contain information about the adjacency measure and symmetry.

The adjacency of the regions is found by using an exoskeleton around each region like proposed in [Fuchs 1998]. Due to the different image partition algorithm, in general we have a complete partition of the image. In this case, we would only need to follow the edges that divide the regions to find the neighbored regions. Due to the over-segmentation of the image partition there exist many small regions. They appear often along weak edges, for example illumination changes. If we take the directly neighbored regions into the adjacency graph only, irrelevant neighborhood information would drop the evidence in the learning step. Deleting these small regions has the difficulty to define an appropriate threshold for the size without canceling useful regions like e.g. roof windows. Small but elongated regions can be of significant size.

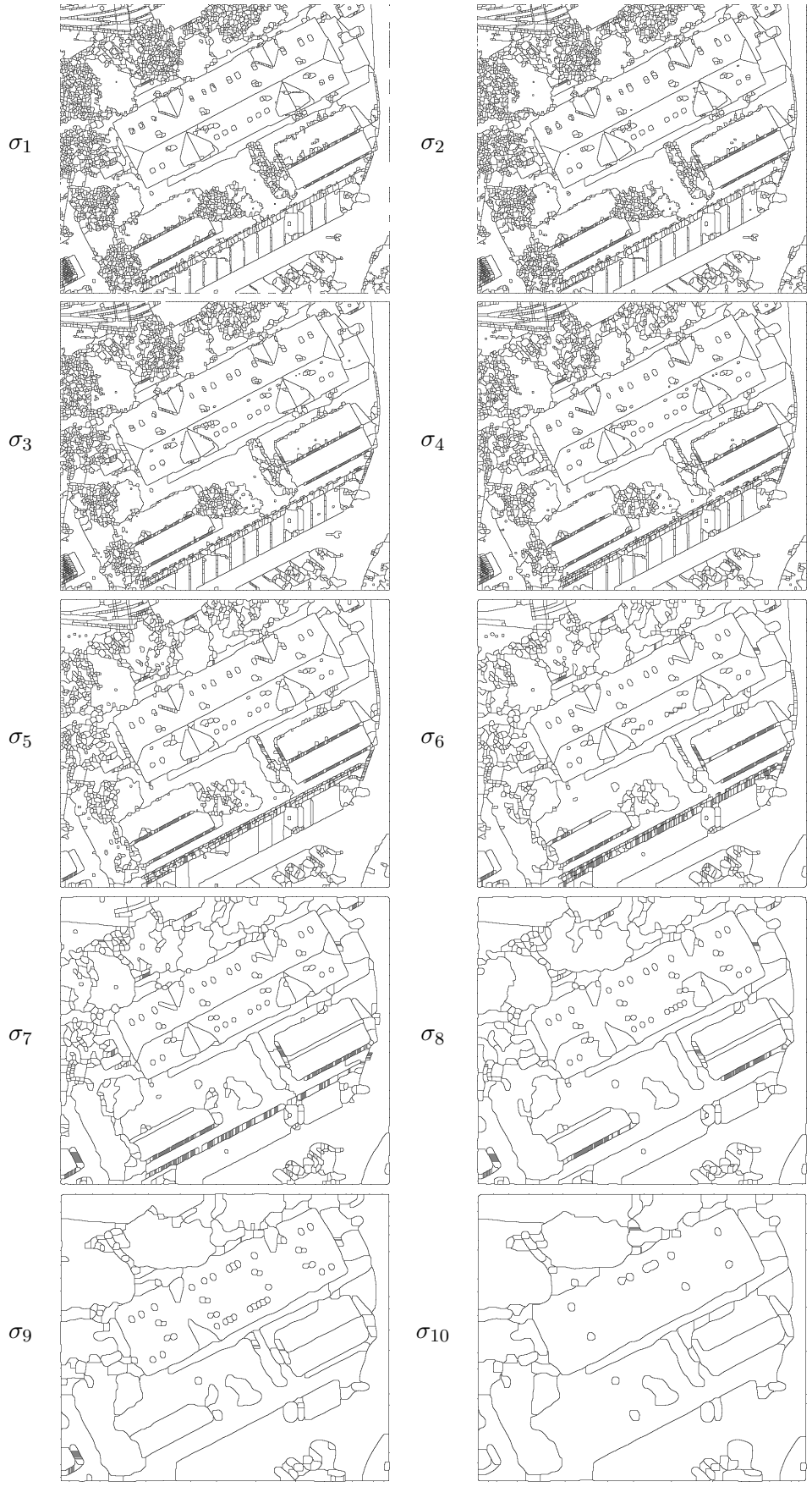


Figure 4.4: The region extraction algorithm on a Gaussian scale space.

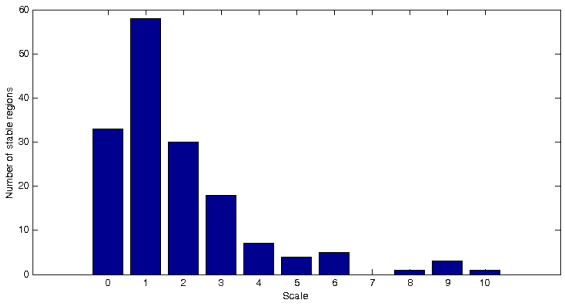


Figure 4.5: The lifespan of the fist 160 regions of the scene in fig. 4.4

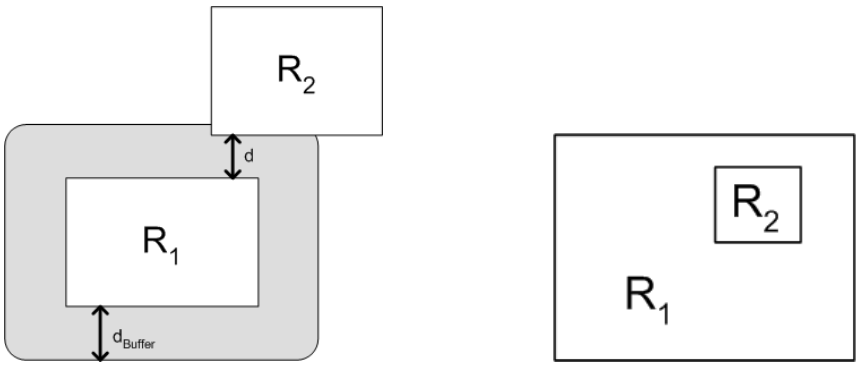


Figure 4.6: Definition of the adjacent regions R_1 and R_2 with a buffer. The distance d between the features is determined via a exoskeleton, the maximal distance is the size of the buffer d_{Buffer}. The inner region on the left has only one neighbor, the outer region has a special attribute for the child region.

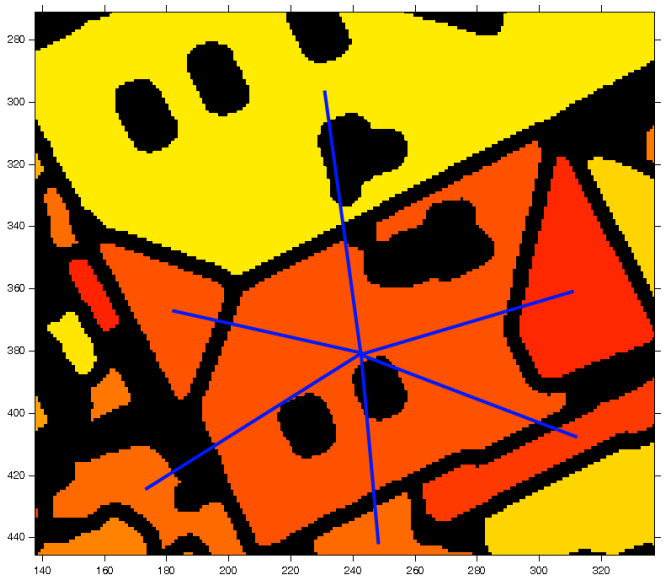


Figure 4.7: The region adjacency graph. The lines between the region denote the adjacency in the graph. Each of these lines has a parameter vector attached, representing the neighbor observations. The end points of the lines in the figure are representative points of their regions.

The goal is to identify every region as adjacent that is contained in a buffer region around the examined region, i.e. the distance to the examined region is below a threshold, like shown in fig. 4.6. The maximum buffer width is set to $d_{max} = 1,8[m]$. The regions that are directly neighbored are addressed as fist-order neighbors later, the regions inside the buffer are second-order neighbors in turn.

We do not extend the adjacency graph over the scale space. This would result in a three dimensional graph. Using this graph as observations in the Bayes net is very complex. In experiments there was no significant increase of the performance with including the scale space information other than the stable lifespan of an object. This is why we implemented the information about the stability as a region observation.

On top of the region adjacency we build a layer that represents all possible cliques of the adjacent regions. This is a preprocessing step for searching the optimal configuration of regions for the observation in the Bayes net. Every clique that can be built out of the adjacent regions is inserted as vetex connected with its region vertices. An edge between the clique vertices is inserted, if the cliques have adjacent but no overlapping regions.

Feature	Name
F_{R1}	Color L
F_{R2}	Color a
F_{R3}	Color b
F_{R4}	Length of boundary
F_{R5}	Size
F_{R6}	Roundness
F_{R7}	Compactness
F_{R8}	2. Moment
F_{R9}	3. Moment
F_{R10}	4. Moment
F_{R11}	Number of Lines
F_{R12}	Number of parallel Lines
F_{R13}	Number of orthogonal Lines
F_{R14}	Number of symmetric Lines
F_{R15}	Number of Neighbors
F_{R16}	Number of equal sized neighbors
F_{R17}	Number of contained neighbors

Table 4.1: List of the used features that are extracted per region in the image processing part.

4.5 Representation of a region

The feature extraction process is region centered. That means that all features and properties are related to the regions.

The first ten features are modeled as continuous variables (Fig. 4.1). For the introduction in the Bayes net we used a quantization table to fit these in the Multinomial distributions. For the color we chose the Lab color space to enable a better clustering in the color space. The length of the outline can be directly observed by counting the boundary pixels. A coding scheme for the boundary length in an eight-pixel neighborhood according to *[Jähne 1989]* is used. Also the size is easy to extract. The roundness is defined as $R \approx \frac{4F}{\pi D^2}$ where F is the size and D the maximal diameter. The compactness is $C \approx \frac{U^2}{F}$, with U representing the outline. The second moment defines the elongation, the third moment the skewness of the region and the fourth moment defines the centrality of the mass. The second and higher moments refer to the major axis of the region.

The eleventh to the 17th feature are modeled as discrete variables. The number of lines and neighbors is small and represents counts in natural numbers that are easier to model with discrete bins in a multinomial distribution. Therefore the number of lines

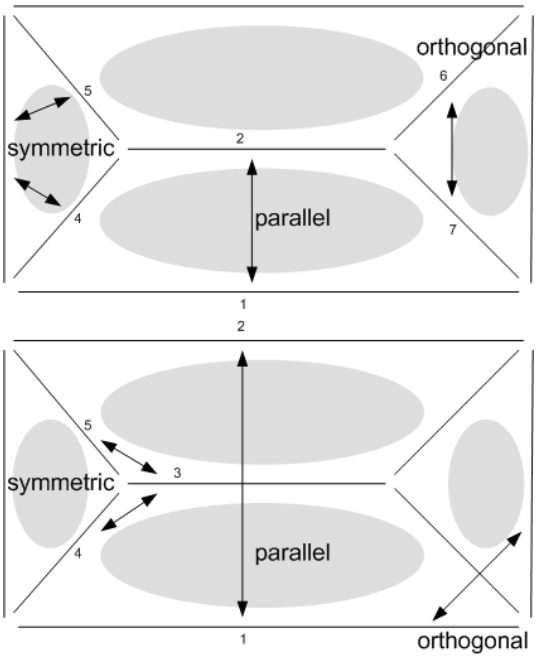


Figure 4.8: This figure explains how the observations of parallel, orthogonal and symmetric lines are done. On the left there are examples for the observations that are done per region on a typical roof layout (hip roof). On the right observations between two adjacent regions are extracted on the same region configuration.

is limited to $(0 \dots 10)$. For the number of lines we count the stratified lines that come out of the Peucker algorithm. According to the filter in the feature extraction step, we omit lines shorter than 25 px. Next, it is checked how many of the lines of the above set are parallel and orthogonal. For symmetry we check that two lines have the same angle with respect to a third line, i.e. the angles have to be equal and the lines have to be connected with a common third line. The configuration is shown in figure 4.8. The angles are allowed to vary ± 3 degree.

The features F_{R15}, F_{R16} and F_{R17} are extracted out of the feature adjacency graph. The number of neighbors is filtered for size. Those which are of the same size $\pm 10\%$ are counted for F_{R16}. The special case of neighbors is the contained neighbor. This appears e.g. if there are windows in the roof or cars on the street. These are counted separately in F_{R17}.

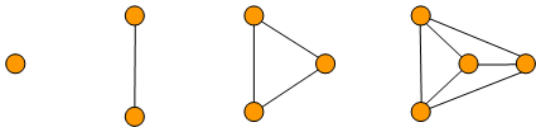


Figure 4.9: Different cliques with increasing number of vertices. Higher ordered cliques do not appear in the planar region graph.

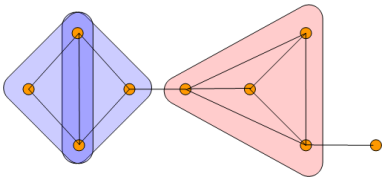


Figure 4.10: A Graph with 9 one-cliques (the vertices), 13 two-cliques (the edges), two 3-cliques (blue) and a 4-clique (red).

4.6 Representation of a neighbor relation

As mentioned before also the edges of the adjacency graph are attributed. The attributes of the neighbor relations are also listed in a vector and contain symmetry information of the regions. The following features are extracted out of the graph:

Features	
F_{N1}	Number of parallel Lines
F_{N2}	Number of orthogonal Lines
F_{N3}	Number of symmetric Lines
F_{N4}	Distance
F_{N5}	Merging

The number of orthogonal and parallel lines is extracted in a similar way like inside the region. The symmetry has to be with respect to a shared line between the two regions like shown in fig. 4.8.

The distance (feature F_{N4}) denotes the minimum distance between the two regions. The merging feature (F_{N5}) represents the scale step from which on the two regions merge.

4.7 Representation of cliques and their adjacency

As a preprocessing step for traversing through every possible combination of cliques we create a layer of vertices on top of the region adjacency graph. Here we create for every clique we instanciate a vertex connected with the connected regions. All pairs of

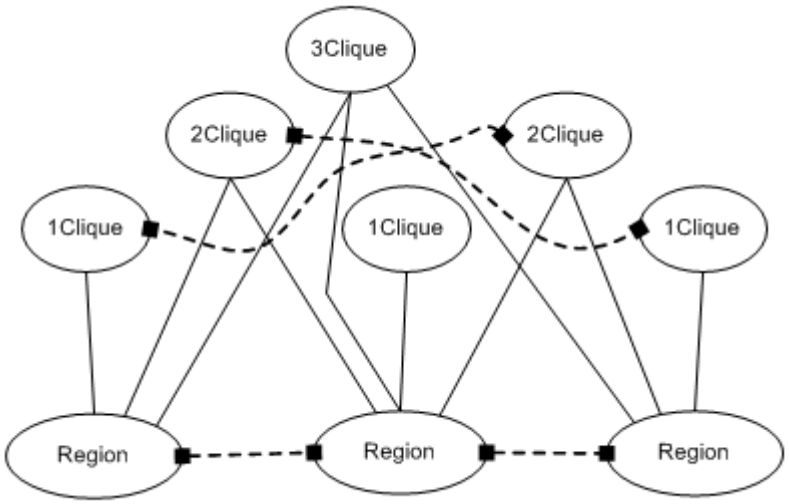


Figure 4.11: The layout of the hierarchic feature graph. The lines represent the affiliation of the image feature regions to the cliques. The dottet lines show adjacency information in the image domain.

clique vertices, that have assigned non overlapping but adjacent regions are connected with a connection marking this neighborhood. This helps to quickly traverse the graph later. For finding combinations of cliques for the object node in the interpretation step (section 5.6 it is necessary to check that also objects with more than two cliques have no overlapping regions assigned to avoid double observations in the Bayes net.

4.8 Conclusion

We have shown in this chapter how we extract certain image features out of the given image. We introduce an attributed feature graph that stores these image features. This graph is parsed in the next chapters and its data is used as observation for the Bayes net.

Chapter 5

Model of the Bayes net and processing

After the extraction of the region graph and its features, the next step is the interpretation of this data. The goal is to obtain an interpretation for every region in the image that aligns with the proposed image model (Chapter 4). To do this we will use a Bayes net due to the motivation above. In the following chapter we will explain the design of the Bayes net that is able to handle the extracted image description.

In the previous chapter we have handled the two dimensions of the image plane and the third dimension of the scale space. Now there is one more dimension to handle with: the semantic ontology.

Our detection method works in a certain small band of spatial resolution. The lower bound of scale of the objects to detect is physically restricted by the image resolution. The upper bound is defined by the search for the stable regions in scale space. It is not the goal to get higher semantic aggregations of objects, e. g. building blocks or the street network. Instead we will use a scene node on top of the Bayes net which represents the type of the whole scene.

5.1 Observing spatial relations with the Bayes net

The concept of Bayes nets is a pure statistical concept in which we need to implement knowledge about the spatial relations of real world objects as well as the relations of their image regions. The spatial relations are introduced implicitly over the region graph and its region cliques. We take the region graph and examine every region together with its relations to neighbor regions. According to our image model, real world objects in

images consist of one or several image regions that have spatial relations to each other. The Bayes net is not aware of spatial relations. We have to introduce groups of regions for examination. This is done by introducing the cliques of regions in the region graph. Cliques in the planar region graph are restricted in their size. That is why each object in the model consists of one or more region cliques.

The image object, i.e. the real world object projected to the image, exists not alone in the image domain. It is embedded in a planar object graph similar to the one on the region level. The graph edges can be equipped with weights that formulate the statistical evidence of existence of an object given its neighbor image objects. Other researchers use a Markov Random Field at this point (e.g. *[Meidow 2000]*). A major problem for the formulation as a Markov Random Field is the normalization of the nodes with an a priori not known and irregular shaped graph structure. Because of this we keep the structure of the Bayes net to model the dependencies between image object nodes. Therefore we introduce a dependent node for every object in the object domain. In this way we model the statistical dependencies of objects in the context of neighbored objects without explicitly introducing spatial relations on the object level. Using this model, we have one Bayes net per object in the image.

5.2 The structure of the Bayes net

The initial model introduces four different levels of nodes like shown in plate writing in figure 5.1. The type and interpretation of nodes will be explained in the following section.

5.2.1 The scene node

The scene node on the top level of the net models the type of the scene in the underlying image. The scene node is a discrete node. Its cardinality depends on the scene types that are introduced. For the case of aerial images these are *urban area*, *suburban area* and *rural area*. For other than middle European airborne imagery other scene types can be introduced. The Bayes node is represented by a multinomial distribution. It is initialized by its prior distribution node Σ that has been noted at the left using a dotted square to show the nature of prior information. The node in the figure is shaded because the node is observed in the learning step.

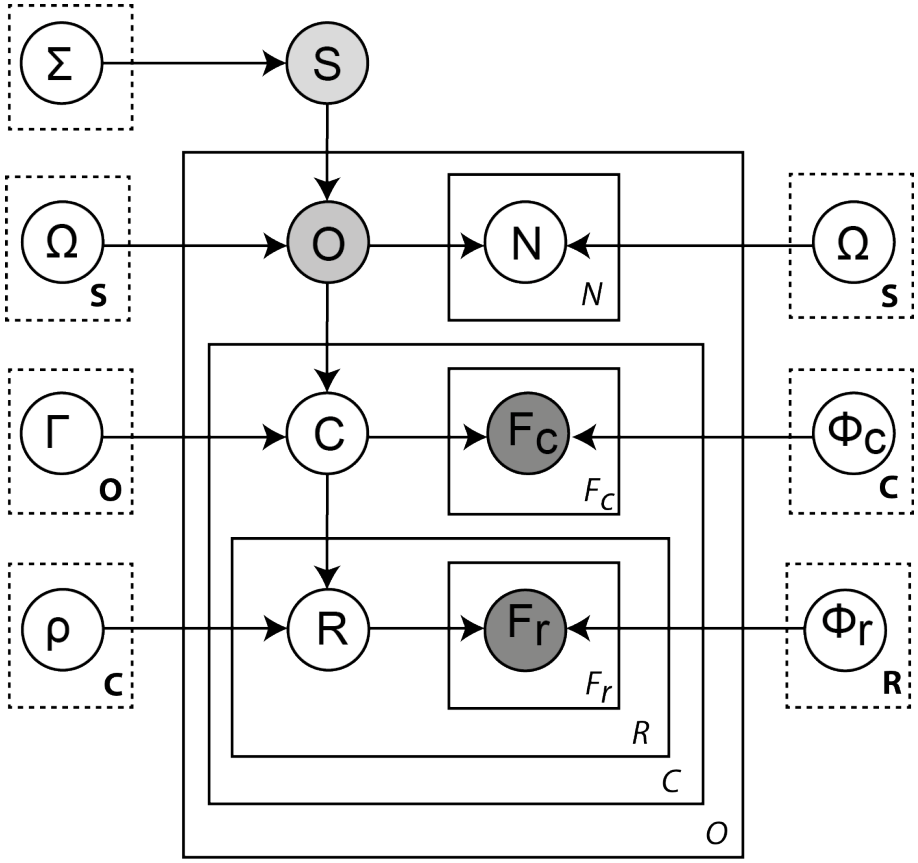


Figure 5.1: The generative model in plate writing. The dark shaded nodes are observed in the detection and the learning step. The light gray shaded nodes are observed during the learning step. The unobserved nodes are not shaded. The nodes at the left denote the prior distributions.

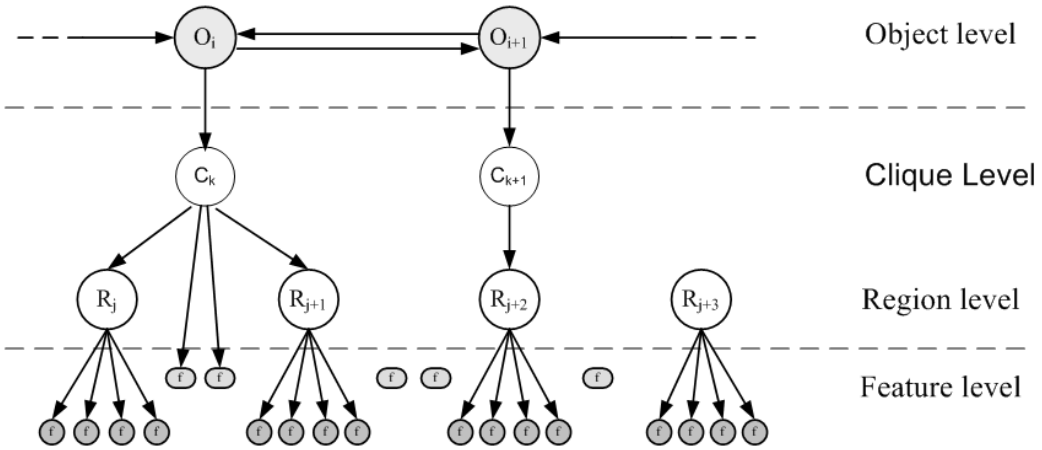


Figure 5.2: The scheme of detection visualized in 2D. The Bayes net is instantiated on top of the planar feature adjacency graph. It uses observations per region and also of the neighbor structure. The objects refer to a dynamic number of cliques and regions.

5.2.2 The object node

A scene can contain many objects, each represented by an object node that is dependent on the scene type node. This node is represented by a discrete random variable $\underline{O}_o$ for the objects $o = 1 \dots O$. The random variable is chosen as a multinomial distribution to model the discrete states of the node, i.e. the object type. Its prior is modeled as dirichlet distribution Ω. The object node is shaded because it is observed in the ground truth database within the learning process. The instances of the object types depend on the data we learn. Examples for the object types are listed in the next chapter.

The object nodes have a strong interdependency with their neighbor objects, therefore they have one or more dependency links to neighbor nodes $\underline{N}$. This node is an object node itself when it is in focus. The dependence to the neighbored objects is in general undirected. The directed arc and with it the dependent modeling is chosen to keep the net structure in a tree structure that has computationally advantages like explained above.

5.2.3 The clique node

Each of the objects decomposes in one or more cliques that are represented by a clique node $\underline{C}_c$. This represents a clique of adjacent image patches that is embedded in the feature graph. The maximum number of cliques is determined in the learning step. This number determines the maximum complexity of buildings for the interpretation step.

The clique node is again a discrete random variable instantiating an inherent vocabulary of clique types. The cardinality of the node is fixed to five (4 for the clique types and one for the state "not existent"). Due to its discreteness the distribution is multinomial. The initialization is a prior distribution that is a Dirichlet distribution Γ.

The modeled cliques range from a one-clique to a four-clique. To avoid getting a different net structure of the Bayes net for the different numbers of dependent regions in the case of different cliques, we model the Bayes net to have always four regions per clique. The region nodes are ordered from one to four. When we have observations of regions, these are introduced starting with node number one.

5.2.4 The region node

As mentioned, the clique node contains one to four region nodes $\underline{R}$. These represent the extracted homogeneous regions in the region graph. The regions are shown in white because they are not observed in the region extraction process for the learning step. The names for the regions are found as "visual words" during the learning step. The region nodes are initialized according to their prior distribution ρ.

5.2.5 The feature nodes

As the lowest level there is the level of feature observations. These features are represented by feature nodes F_i with $i \in 1..17$ for the region features and with $i \in 18..21$ for the clique features respectively. The number of features is fixed and given by the feature extraction part of the system. In our feature extraction we have modeled 17 region features and six different observations for the cliques. The type of extracted features are explained in chapter 4. The features are modeled by different distributions with respect to their different parameter spaces. The variables of the Bayes net are modeled also as discrete distributions. For float values we use a quantization to introduce them to the discrete classes. All prior distributions are modeled as vectors of parameters. The size of the vectors is given in bold face letters in the squares of the prior node and is set to the cardinality of the parent node.

5.3 Sampling the net

The model is easy to understand if we examine how it can be used to generate a synthetic dataset by sampling the Bayes net. Therefore we start at the top node and follow the causal dependences:

- The scene node represents the category of the image scene. By drawing this variable, its outcome will decide about the change of all dependent nodes. The variables distribution is

$$\underline{S} \sim \text{Multin}(S \mid \Sigma) \tag{5.1}$$

with the prior distribution Σ.

- Given the scene type we generate objects. The number of objects per scene is not modeled, so the model cannot give any information at this stage for generating synthetic objects. The information about the objects geometry is generated further down in the feature nodes. This deficiency is discussed later in chapter 7. The objects are generated until the scene is filled with objects. The object node is chosen according to

$$\underline{O} \sim \text{Multin}(O \mid S, \Omega_S) \tag{5.2}$$

where the distribution of S is chosen according to the scene category and the neighbor node types.

- The dependence of the neighbor nodes N has to be considered in the following objects. The dependency on the neighbor objects is given in both ways. The neighbor nodes are again object nodes that represent objects which are adjacent in the scene to the surveyed one. For sampling the net we do not have to draw samples from the neighbor node. It is represented by the other (neighbored) object nodes.
- The clique node depends only on the object type and can be generated according to

$$\underline{C} \sim \text{Multin}(C \mid O, \Gamma_C). \tag{5.3}$$

The number of cliques is unknown in the beginning. We instantiate all four clique nodes. In the case that we sample the clique type "'not existent"' we stop to sample further down the tree.

- The region is chosen according to

$$\underline{R} \sim \text{Multin}(P \mid C, \rho_P) \tag{5.4}$$

- On the lowest level the feature nodes are used to generate values for the image

features. These feature nodes represent the features listed in 4.1.

$$\underline{F_i} \sim \text{Multin}(F_i \mid R, \phi_i) \tag{5.5}$$

This is equivalently done with the feature nodes under the clique node representing the relations between the regions $\underline{F_c}$.

5.4 Joint distribution and priors

According to the model, the Bayes net is organized in a tree form and is modeled only with discrete random variables. This is the most effective form for learning and inference in a Bayes net. The whole Bayes net models a joint distribution $P(S, O, C, R, F \mid \Sigma, \Omega, \Gamma, \rho, \phi)$ that factorizes in form of

$$\begin{aligned} P\left(S, O_m, C_n, R_l, F_i \mid \Sigma, \Omega, \Gamma, \rho, \phi\right) = & P(\Sigma)P(S \mid \Sigma) \\ & \left(\sum_O P(\Omega_S)P(O \mid \Omega_S) \sum_N P(N \mid O)\right. \\ & \left(\sum_C P(\Gamma_O)P(C \mid O, \Gamma_O)P(F_C \mid C)\right. \\ & \left.\left.\left(\sum_R P(\rho)P(R \mid C, \rho)P(\phi)P(F_R \mid R, \phi)\right)\right)\right) \end{aligned} \tag{5.6}$$

5.5 Learning method

After defining the general layout of the Bayes net, we can now begin to determine the two parts that are not fixed in the Bayes net yet: the probability distribution and the structure of the net in the clique and region nodes.

If every node would be annotated, the learning task in Bayes nets would reduce to counting the cases. In the detection problem that is presented here, the problem of learning is more complex. We have two issues based on the kind of observations that we have to handle while learning the Bayes net.

First we do not have fully observed data because it happens that the feature extraction is not able to extract every region correctly. This case can occur due to occlusions, image distortions or just a wrong modeled image noise and leads to missing or wrong regions in the region adjacency graph. The low level features that are attached to the regions,

i.e. the observations per region, are complete.

Second, the labels for the region and clique nodes are not given a priori. Also these nodes are not observed. The region and clique nodes are not contained in the dataset that is provided for learning, so we do not know the instances and the number of their labels. It is a task of the learning algorithm to create labels. This is modeled as a visual words vocabulary and leads to an unknown cardinality of the discrete distribution of the region node. This unknown cardinality is equivalent to an unknown structure with the restriction that we know where the dependencies in the Bayes net are.

We are using the Structural EM-algorithm according to section 3.5 for the learning step. In each iteration not only the parameters but also the graph of the Bayes net is changed to find an optimal set. The change of the graph in our case means changing the number of clique and region nodes. With that we can create labels of the clique and region nodes. The introduction of new nodes (i.e. new labels) increases the acceptance of the net, so the algorithm tends to introduce as many nodes as possible, leading to an overfitting of the model. To avoid this overfitting in this estimation, we use a BIC-term for punishing the creation of new states.

The learning is carried out on every object $O \in \mathcal{O}$ in the groundtruth dataset. Ground truth is provided in form of annotated images of an image database. The annotated image database is kept in format of the LabelMe-Database *[Russell et al. 2005]* of the MIT-group. The annotation consists of simple polygons for the objects with no relation given between the different items. The annotation polygons are taken around the objects that the borders are kept inside the polygons. There is no given ontology inside the LabelMe toolbox. This was provided in form of an acquisition rule in which defines the order of the object and its name. The annotation was then picked by hand by operators using a tool shown in figure 5.3.

Misinterpretations of the image and the acquisition rules must be considered by using this data, so the training data set is not error free.

In the aerial database there are image parts that are cut out of aerial images. The images belong to four different flights that are explained in the next chapter. Because of the size of aerial images and the memory limitations in MATLAB the images are cut into tiles of 1000 by 1000 pixels. The cutting was done without measuring the distance and orientation to the nadir point, so the information that would be useful for calculating the radial distortion was lost.

Since the automatic feature extraction of image regions differs from the human extraction of the ground truth data set, we have to handle these differences. The existence of a region is introduced by comparing the extracted regions with those polygons stored

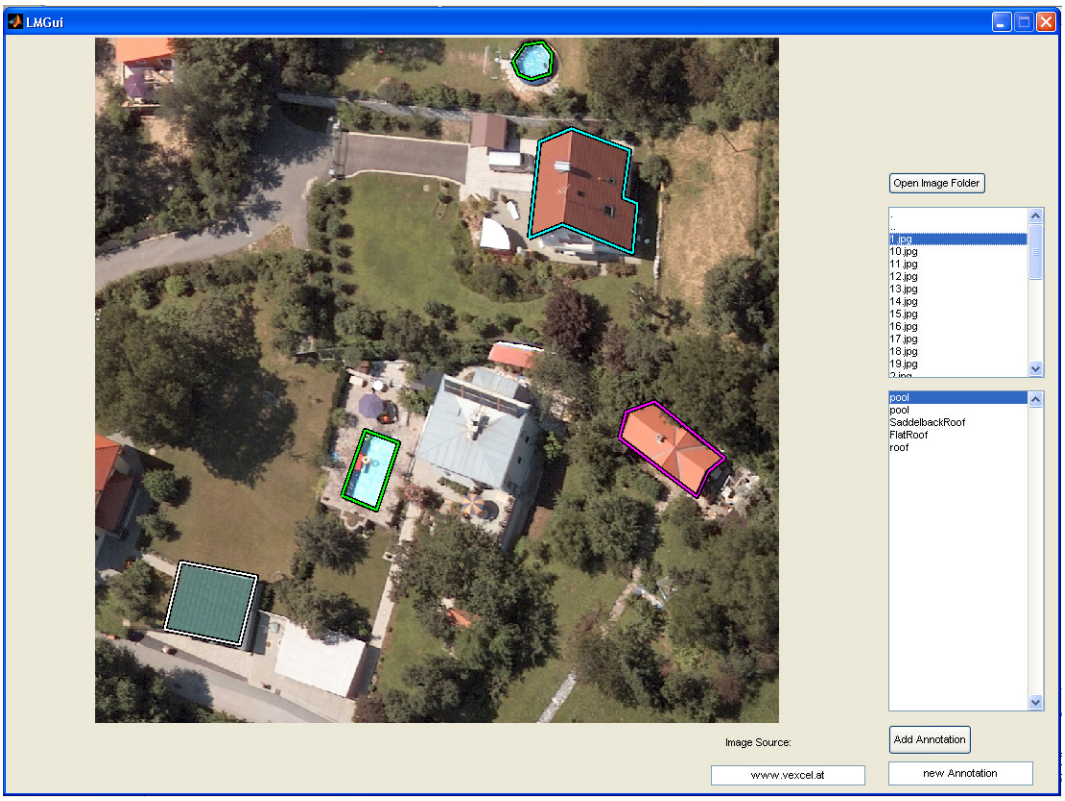


Figure 5.3: The GUI for classification of the groundtruth data. In this screenshot five objects are captured: two pools, a saddleback roof, a flat roof and a not categorized roof.

in the ground truth database. By thresholding the overlap of the segmented region with the polygon consensus is found. Here we introduced an overlap of minimum 75% of the region area for a hit. If an automatically extracted region overlaps less, it will be discarded.

5.6 Interpretation method

The goal of the detection is to label each region according to its object type and thus to interpret the scene. In the detection step the Bayes net including the structure and the parameters for steering the probabilities are given. The feature extraction supplies the algorithm with the hierarchical region graph that is used to introduce the feature observations. We search for any constellation that maximizes the probability for the observed features $p(\mathcal{F} \mid \mathcal{M}, \Theta)$ over the whole scene.

We start with creating nodes for all regions that are found in the image feature

extraction part. For every region we introduce the observed features nodes for all image features of the region as described in the last chapter. The task is now to find the optimal combination of regions to infer the most probable clique and object nodes. So we take recombinations of the region nodes to build a Bayes net on top of them with the type of net topology we have learned in the learning step before. For the clique nodes we have to search for adjacent image regions that form a clique; for the object nodes, we search then for adjacent clique nodes. For this search we can use again the hierarchical image feature graph we have presented in chapter 4. On top of the graph vertices that represent the regions, we have built a layer where every possible clique combination is represented. Between these vertices we have introduced an edge as adjacency information that shows if two cliques have adjacent but no overlapping regions. This information we use to iterate through the possible configurations for observations in the Bayes net. We implement and connect the clique nodes according to the vertices in the clique layer and do the same with the object nodes. For them the adjacency link of the clique vertices is important, but it is necessary to check for double observations of regions for objects with more than two regions.

When we model the assignment of the observed features to a Bayes net as $\underline{a}$, we search for

$$a' = \underset{a}{\operatorname{argmax}}\ P(\mathcal{F} \mid \mathcal{M}, \Theta, a).$$

As we have seen above, the search space for a is restricted due to the region graph. We can only introduce observations of regions, cliques and objects to a Bayes net that are implemented in the feature adjacency graph.

We use the message passing inference algorithm according to *Kevin P Murphy [2012]* to propagate the knowledge to the upper nodes of the net. The propagation cannot be done straight forward. The object nodes depend on their neighbor nodes. That means we have to iterate The labeling is done by evaluating the MAP estimation throughout all levels of the Bayes net.

5.7 Conclusion

In this chapter we introduced the structure of a Bayes net which can be used to observe spatial relations in the extracted image feature graph. We have shown the details of the net concerning the interpretation of the nodes of the Bayes net and their distributions and priors. Also we have shown how this Bayes net can be learned by feeding training data to it and how to to interpret aerial images with this prelearned knowledge base.

Chapter 6

Experiments and results

In this chapter we will show that the model that we developed in the previous chapters holds and can be used to learn and interpret real world data. To do this, we will apply the interpretation to data sets with different properties. Additionally we will provide several modifications to the Bayes net and the interpretation scheme and will look at the results to explain how the model reacts in detail.

6.1 The ground truth

For all experiments we use four different datasets that were organized like shown in figure 6.1 to learn and detect objects in images. The datasets that are available are:

Graz-Andritz (GRA) The dataset of Graz-Andritz consists of aerial images that are taken with a digital aerial multispectral camera in the region around Graz, Austria (figure 6.1). Here, only the RGB-spectral parts of visible light are used. The images contain much vegetation and trees around the buildings. The buildings are often solitary houses. Row houses and industrial buildings are the minority. All classical roof forms are available, but the classical and quite complicated roof forms are common, e.g. the half-hip roof. Also dormers are common on many buildings. In addition there are many streets, parking lots and cars labeled in these images. Some houses have swimming pools in their garden that have been labeled as an extra class. The scene is classified as a rural scene.

Graz-Centrum (GRZ) The Graz-Centrum dataset consists of images of the same camera like above. The images are taken of down-town Graz. The camera is slightly defocused, so the image seems to be smoothed at some locations. The

Figure 6.1: Typical images from the dataset Graz-Andritz. It shows the landscape around Graz, which is partly rural, partly suburban.

objects in the images are mostly row houses and streets. The roof types are often saddle roofs along the streets, but complicated roof types are common with newer buildings. The roofs are often old and highly textured due to moss and age. This leads to a heavy over-segmentation.

Vegetation is barely shown; some trees exist in the streets and on squares. Cars and asphalt are very common, in contrast to the Graz-Andritz dataset streets are not only elongated structures. Many backyards and squares are used for parking. There are many specialties like sunshades on the market that have not been labeled in the ground truth database, so that these will not be known in the detection. The scene label is denoted as urban scene.

Bonn-Ippendorf (BOI) Ippendorf is a district of Bonn, Germany, that has suburban structures. The images are taken with an analog camera for the purpose of mapping. The images show a much higher noise and the scale is smaller than the digital images of Graz. The roof structures are in majority saddle roofs. There are some larger buildings that have flat roofs and rarely there are solitary houses with walm-roofs. There are green backyards that are typical for suburban scenes. This scene is labeled as suburban scene.

Toyonaka (TOY) This dataset shows the down-town area of Toyonaka, Japan. The

Figure 6.2: Typical images from the dataset Graz Centrum, a dense urban scene of the down town area of Graz.

Figure 6.3: Typical images from the dataset Bonn-Ippendorf. Ippendorf is a typical german suburban area with many semi detached buildings and row houses along the streets, interrupted sometimes by bigger units.

Figure 6.4: Typical images from the dataset Toyonaka (Japan).

images consist of scenes that are not found in European scenery: The buildings are very close together and the roof colors are diverse. In contrast to European roof types, these are quite nested and there exist yellow, red, blue and green colors for the roof. Next to the small and uniform private buildings exist huge flat top buildings. The specialty of this set of images is, that these were taken with a very high overlap between the images. This leads to the fact, that every building is shown on up to nine images from different perspectives. The illumination of this dataset is very diffuse. There are no real shadows; nearly every detail on the ground, also between the buildings is visible. Also this dataset has the label urban scene which can be introduced as observation for the scene node.

These datasets are used for experiments in the following sections. In the aerial database there are approximately 200 image parts that are cut out of 60 aerial images. The images belong to four different flights that are described before. In the database are approximately 2100 labeled objects containing 1300 buildings of any type.

All datasets are not completely labeled, i.e. there are only salient objects that have a label. The objects, that are completely incorporated in the datasets are the three building (roof) types and the streets. The object types car, vegetation, grass, shadow and pool are only partly labeled. Objects that are truncated at the image border are not labeled either. This property of the ground truth dataset has to be kept in mind when evaluating the detection results. This is why there are some results that could be

Number	Types	
210	Flat roof building	[frb]
465	Saddle roof building	[srb]
579	Other roof building	[orb]
159	Street area	[str]
180	Car	[car]
30	Pool	[poo]
91	Grass	[gra]
199	Vegetation	[veg]
258	Shadow	[sha]

Table 6.1: The table shows the number of labeled objects contained in the ground truth database.

interpreted as false positives, which are not.

6.2 Results of the detection

For evaluating the performance of the interpretation, we make several experiments with different parameter sets and images. First, we like to proove how good the interpretation task works under standard working conditions, i.e. to interpret scenes of the same type like learned before. Therefore we divide each of the data sets into two parts: One is used for learning, the second half is used to evaluate the detection. During the learning step we create the parameter set for each of the five datasets. These are applied to the second half of each of the data set. The results are shown in fig. 6.5(a) to 6.5(d) as confusion matrices of the reference classes versus the result classes. The correct classifications are shown on the main diagonal. The off diagonal elements represent the percentage of classifications that were not correctly identified. The colors of the elements are chosen on a linear color palette from green (0%) to red (100%) to help the quick visual interpretation of the results.

In the Graz-Andritz dataset [GRA] 6.5(a) we have strong support for the three building classes. The misclassifications of the buildings are interpreted as one of the other building classes. Only the saddle roof building class is interpreted in a low percentage as vegetation and shadow. For the class of the cars it it noticable that they were not distinguishable enough, so the rejection class was taken in 18% of the cars in the reference dataset. Other high rejection rates are among the classes of the shadow and the pools. The shadow is also often (18%) misclassified as vegetation. An explanation for this is, that the vegetation class and the shadow class point to irregular shaped regions if the

shadows origin is itself vegetation e.g. a tree. Also the grass regions are misclassified as vegetation. This shows that the geometric and color observations are not good enough for a strong interpretation of these classes.

In the dataset of down town area of Graz 6.5(b) we see the same behaviour among the build classes. The correctness of the class of cars is only at 60%. This could be the outcome of many objects of similar size, shape and color on the streets of Graz, e.g. the often occuring sunshades. Forthermore we see a high rejection percetage in the class for grass. Since in the down town scene there are not many grass labels, this is not enough to imprint the probability distribution of the grass node of the Bayes net.

The dataset of Bonn 6.5(c) shows a misclassification rate of 31.2% for the class of *other roof buidings* as *saddle roof building*. This happens due to the similarity of some complex roofs of individual building to the saddle roof class.

The buildings and streets in the Toyonaka dataset 6.5(d) have a very different setup. Since the roofs are often blue, green or white, we have the proof, that the classification does not only rely on the color observation. The correctness of the classification is between 80% and 90% and is compareable to the other datasets. Since there is nealy no vegetation or grass visible on the images, these classes are not very strong. The grass regions are labeld as vegetation for 100% and the vegetation regions were only fit with 50% correctness. The aerial image is taken in a very diffuse light, so the shadows only appear in the narrow space between buildings. This is rejected in a big percentage (25.7%).

We can observe that the performance is quite equal for each of the data sets. There are some misclassifications among the roof classes, also the grass- and vegetation class have some deficiency in selectivity.

As a next step we vary the threshold for the rejection class. If the threshold is set to Zero, the interpretation is carried out according to the maximum a posteriori estimation. Every region is labeled according to the maximum (a posteriori) probability. Therefore no rejection class is possible. With a threshold above zero the rejection class is chosen, if the distance between the two highest probabilities is below a threshold. This can happen, if there is a kind of uniform distribution in the probability vector, i.e. multiple object classes have the same likelihood. We introduce thresholds of 10%, 20%, 30% and 40%. The results are shown in fig. 6.9. We observe that the number of rejected objects increases from the beginning while the correct interpretation begins to decrease over a threshold of 20%. Applying the threshold helps to increase the sensitivity since the weak classifications are rejected. Stronger thresholds reject also correct interpretations.

In another experiment we learn the parameters using several data sets and examine

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	pol	rej
srb	81,6	4,1	8,2	0,0	0,0	0,0	2,0	2,0	0,0	2,0
frb	4,8	90,5	4,8	0,0	0,0	0,0	0,0	0,0	0,0	0,0
orb	8,5	1,7	83,1	0,0	0,0	1,7	0,0	3,4	0,0	1,7
car	0,0	0,0	1,8	74,5	0,0	1,8	1,8	1,8	0,0	18,2
str	0,0	3,4	0,0	0,0	79,3	0,0	3,4	13,8	0,0	0,0
gra	0,0	2,3	0,0	0,0	0,0	67,4	23,3	0,0	0,0	7,0
veg	0,0	0,0	0,0	1,8	0,0	5,4	87,5	1,8	0,0	3,6
sha	0,0	0,0	0,0	0,0	0,0	5,4	18,9	56,8	0,0	18,9
pol	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	83,3	16,7

(a) Confusion Matrix for the dataset Graz-Andritz.

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	rej
srb	90,1	2,8	2,8	0,0	0,0	0,0	0,0	0,0	4,2
frb	3,8	76,9	7,7	0,0	0,0	3,8	0,0	0,0	7,7
orb	2,9	1,4	88,4	0,0	0,0	0,0	0,0	2,9	4,3
car	0,0	0,0	0,0	60,0	0,0	0,0	0,0	20,0	20,0
str	0,0	0,0	7,7	0,0	76,9	0,0	0,0	15,4	0,0
gra	0,0	0,0	0,0	0,0	0,0	16,7	0,0	0,0	83,3
veg	0,0	0,0	0,0	0,0	0,0	14,3	85,7	0,0	0,0
sha	0,0	0,0	0,0	0,0	0,0	5,3	5,3	84,2	5,3

(b) Confusion Matrix for the dataset Graz Zentrum.

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	rej
srb	90,2	0,0	3,3	0,0	0,0	0,0	0,0	1,6	4,9
frb	0,0	92,2	2,0	0,0	0,0	0,0	0,0	0,0	5,9
orb	31,3	0,0	59,4	0,0	0,0	0,0	0,0	3,1	6,3
car	0,0	0,0	0,0	76,9	0,0	0,0	7,7	7,7	7,7
str	0,0	0,0	0,0	0,0	87,0	0,0	0,0	8,7	4,3
gra	0,0	0,0	0,0	0,0	0,0	63,6	27,3	0,0	9,1
veg	0,0	0,0	0,0	0,0	0,0	5,9	76,5	5,9	11,8
sha	0,0	0,0	0,0	0,0	2,3	0,0	4,7	81,4	11,6

(c) Confusion Matrix for the dataset Bonn-Ippendorf.

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	rej
srb	80,8	1,0	4,0	0,0	0,0	0,0	0,0	5,1	9,1
frb	3,6	89,3	7,1	0,0	0,0	0,0	0,0	0,0	0,0
orb	8,6	0,0	80,2	0,0	0,0	0,0	0,0	3,2	8,0
car	0,0	0,0	0,0	69,4	0,0	0,0	0,0	11,1	19,4
str	0,0	0,0	0,0	0,0	96,8	0,0	0,0	0,0	3,2
gra	0,0	0,0	0,0	0,0	0,0	0,0	100	0,0	0,0
veg	0,0	0,0	0,0	0,0	0,0	0,0	50,0	0,0	50,0
sha	1,8	1,8	5,4	0,0	1,8	0,0	0,0	53,6	35,7

(d) Confusion Matrix for the dataset Toyonaka.

Figure 6.5: The numbers represent the percentage of the classification. The correct classification is shown on the main diagonal. Every other value represents a misclassification. The last column represents the rejection class which is applied here with a threshold of 20%.

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	pol	rej
srb	81,6	6,1	8,2	0,0	0,0	0,0	2,0	2,0	0,0	0,0
frb	4,8	90,5	4,8	0,0	0,0	0,0	0,0	0,0	0,0	0,0
orb	8,5	1,7	83,1	0,0	0,0	1,7	0,0	3,4	0,0	1,7
car	0,0	0,0	1,8	78,2	0,0	3,6	3,6	3,6	0,0	9,1
str	0,0	3,4	0,0	0,0	79,3	0,0	3,4	13,8	0,0	0,0
gra	0,0	2,3	0,0	0,0	0,0	67,4	27,9	0,0	0,0	2,3
veg	0,0	0,0	0,0	1,8	0,0	7,1	87,5	3,6	0,0	0,0
sha	0,0	0,0	0,0	2,7	2,7	10,8	21,6	56,8	0,0	5,4
pol	0,0	0,0	0,0	5,6	0,0	0,0	0,0	5,6	88,9	0,0

(a) Confusion Matrix for the dataset Graz-Andritz.

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	rej
srb	90,1	2,8	4,2	0,0	0,0	0,0	0,0	1,4	1,4
frb	3,8	76,9	7,7	0,0	0,0	3,8	0,0	3,8	3,8
orb	2,9	2,9	88,4	0,0	0,0	0,0	1,4	2,9	1,4
car	0,0	0,0	0,0	60,0	0,0	0,0	0,0	40,0	0,0
str	0,0	0,0	7,7	0,0	76,9	0,0	0,0	15,4	0,0
gra	0,0	0,0	0,0	0,0	0,0	16,7	0,0	33,3	50,0
veg	0,0	0,0	0,0	0,0	0,0	14,3	85,7	0,0	0,0
sha	0,0	0,0	0,0	0,0	0,0	5,3	5,3	84,2	5,3

(b) Confusion Matrix for the dataset Graz Zentrum.

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	rej
srb	90,2	1,6	4,9	0,0	0,0	0,0	0,0	3,3	0,0
frb	0,0	92,2	2,0	0,0	0,0	0,0	0,0	2,0	3,9
orb	31,3	0,0	59,4	0,0	0,0	0,0	0,0	6,3	3,1
car	0,0	0,0	0,0	76,9	0,0	0,0	7,7	7,7	7,7
str	0,0	0,0	0,0	0,0	87,0	0,0	0,0	8,7	4,3
gra	0,0	0,0	0,0	0,0	0,0	63,6	27,3	0,0	9,1
veg	0,0	0,0	0,0	0,0	2,0	9,8	76,5	5,9	5,9
sha	0,0	0,0	0,0	0,0	4,7	0,0	11,6	81,4	2,3

(c) Confusion Matrix for the dataset Bonn-Ippendorf.

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	rej
srb	80,8	4,0	4,0	0,0	0,0	0,0	0,0	6,1	5,1
frb	3,6	89,3	7,1	0,0	0,0	0,0	0,0	0,0	0,0
orb	9,6	1,1	81,3	0,0	0,0	0,0	0,0	5,3	2,7
car	0,0	0,0	0,0	69,4	0,0	0,0	2,8	13,9	13,9
str	0,0	0,0	0,0	0,0	96,8	0,0	0,0	0,0	3,2
gra	0,0	0,0	0,0	0,0	0,0	0,0	100	0,0	0,0
veg	0,0	0,0	0,0	0,0	0,0	0,0	83,3	0,0	16,7
sha	1,8	1,8	5,4	0,0	3,6	0,0	3,6	71,4	12,5

(d) Confusion Matrix for the dataset Toyonaka.

Figure 6.6: This figure shows the again the percentage of the classification like in fig. 6.5. Here, we apply a threshold of 10%.

the correlation for different data sets. First the two datasets of Graz-Andritz and Graz-Center, and then the data sets of Granz Andritz, Bonn Ippendorf an Graz Center are chosen. The result is a slight decrease of correct interpretation (fig. 6.7(a) and 6.7(b)). At this point we can look also on the interpretation of the scene node. As mentioned before, we associated the classes *rural*, *suburban* and *urban* with the data sets of Graz-Andritz, Bonn Ippendorf and Graz-Zentrum. To investigate the sensitivity of the interpretation with reference to the given label of the scene node. Therefore we intentionally changed the label. The result is shown in table 6.12 . We preset the scene by introducing the scene as observation during the interpretation task. This yields nearly the same interpretation results like the case with subdivided classes with 74% (fig. 6.7(c)) and 78% (fig. 6.7(d)).

In the next modification we investigate the influence of the different feature observations on the interpretation results. Therefore we leave out the color observations and in a second step the neighbor observations.

Leaving out the color observation in the feature vector results for the GA dataset mainly in two areas: The misclassifications between the classes *pool* and *flat roof* increase significantly as well as the misclassification between *vegetation* and *grass*(fig. 6.8(a)). In these classes, the color observation is an important feature that contributes stronlgy to the right classification. Leaving out the neighborhood dependencies results in a general decrease of the performance (fig. 6.8(b)). The main changes here are visible in the *car* and the *pool* class. These benefit mostly from the neighbor information. Missing this information the cars are often classified as vegetation or shadow. The pools are also recognized as flat roof buildings.

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	rej
srb	75,8	3,3	6,7	0,0	0,0	0,8	1,7	4,2	7,5
frb	4,3	70,2	12,8	0,0	2,1	2,1	4,3	0,0	4,3
orb	9,4	0,0	74,2	0,0	0,0	0,0	0,0	4,7	11,7
car	0,0	0,0	0,0	60,0	0,0	0,0	0,0	6,7	33,3
str	0,0	0,0	0,0	0,0	97,6	0,0	0,0	0,0	2,4
gra	0,0	0,0	0,0	0,0	0,0	61,2	20,4	0,0	18,4
veg	0,0	0,0	0,0	0,0	0,0	17,5	65,1	0,0	17,5
sha	0,0	1,8	5,4	0,0	1,8	0,0	0,0	55,4	35,7

(a) Confusion Matrix for the mixed datasets GRA and GRZ.

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	rej
srb	77,3	3,9	5,5	0,0	0,0	0,0	0,0	2,8	10,5
frb	5,1	66,3	10,2	0,0	0,0	0,0	0,0	9,2	9,2
orb	10,0	0,0	68,8	0,0	0,0	0,0	0,0	3,8	17,5
car	0,0	0,0	0,0	75,3	0,0	0,0	0,0	5,5	19,2
str	0,0	0,0	0,0	0,0	92,3	0,0	0,0	6,2	1,5
gra	0,0	0,0	0,0	0,0	0,0	91,7	5,0	0,0	3,3
veg	0,0	0,0	0,0	0,0	0,0	8,8	85,1	3,5	2,6
sha	1,0	1,0	3,0	0,0	1,0	0,0	0,0	66,7	27,3

(b) Confusion Matrix for the mixed datasets GRA, GRZ and BOI.

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	rej
srb	79,2	2,5	4,2	0,0	0,0	0,8	2,5	3,3	7,5
frb	6,4	72,3	8,5	0,0	2,1	2,1	4,3	0,0	4,3
orb	7,0	0,0	82,0	0,0	0,0	0,0	0,0	3,9	7,8
car	0,0	0,0	0,0	61,7	0,0	0,0	1,7	8,3	28,3
str	0,0	0,0	0,0	0,0	97,6	2,4	0,0	0,0	0,0
gra	0,0	0,0	0,0	0,0	0,0	63,3	18,4	2,0	16,3
veg	0,0	0,0	0,0	0,0	1,6	15,9	63,5	0,0	19,0
sha	0,0	0,0	0,0	0,0	1,8	1,8	7,1	71,4	17,9

(c) Confusion Matrix for the same mixed datasets like in fig. 6.7(a) but with an observed scene type.

reference \ detection	srb	frb	orb	car	str	gra	veg	sha	rej
srb	82,9	3,9	5,0	0,0	0,0	0,0	0,0	3,3	5,0
frb	5,1	71,4	10,2	0,0	0,0	0,0	0,0	5,1	8,2
orb	11,9	5,0	68,8	0,0	0,0	0,0	0,0	3,1	11,3
car	0,0	0,0	0,0	74,0	0,0	0,0	1,4	4,1	20,5
str	0,0	0,0	0,0	0,0	90,8	0,0	0,0	7,7	1,5
gra	0,0	0,0	0,0	0,0	0,0	83,3	6,7	3,3	6,7
veg	0,0	0,0	0,0	0,0	0,0	7,9	86,8	3,5	1,8
sha	0,0	0,0	1,0	0,0	1,0	0,0	0,0	74,7	23,2

(d) Confusion Matrix for the same mixed datasets like in fig. 6.7(b) but with an observed scene type.

Figure 6.7: The first two figures show the classification results for two sets of images from different data sources. The detection is carried out like the experiments before. The last two results show increased values. Here we observed also the scene type of each image, which has the effect of a preset in the Bayes Net.

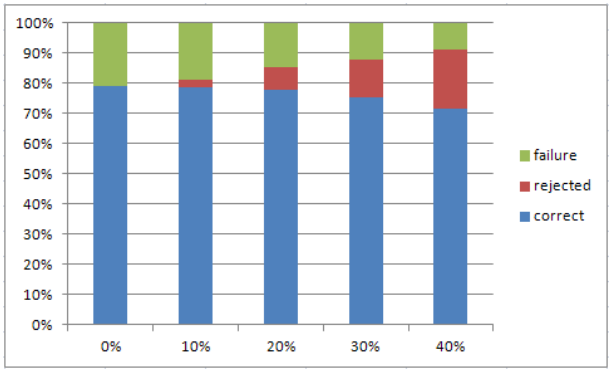
reference \ detection	srb	frb	orb	car	str	gra	veg	sha	pol	rej
srb	83,7	4,1	8,2	0,0	0,0	0,0	0,0	2,0	0,0	2,0
frb	4,8	90,5	4,8	0,0	0,0	0,0	0,0	0,0	0,0	0,0
orb	8,5	3,4	81,4	0,0	0,0	1,7	0,0	1,7	0,0	3,4
car	0,0	0,0	1,8	70,9	0,0	3,6	1,8	3,6	0,0	18,2
str	0,0	0,0	0,0	0,0	82,8	0,0	3,4	13,8	0,0	0,0
gra	0,0	0,0	0,0	0,0	0,0	46,5	25,6	20,9	0,0	7,0
veg	0,0	0,0	0,0	1,8	0,0	32,1	53,6	8,9	0,0	3,6
sha	0,0	0,0	0,0	0,0	0,0	21,6	21,6	37,8	0,0	18,9
pol	0,0	33,3	5,6	0,0	0,0	0,0	0,0	0,0	38,9	22,2

(a) The results of the GRA dataset with a 20% rejection. Here the Bayes net was modified so that the color information was not used.

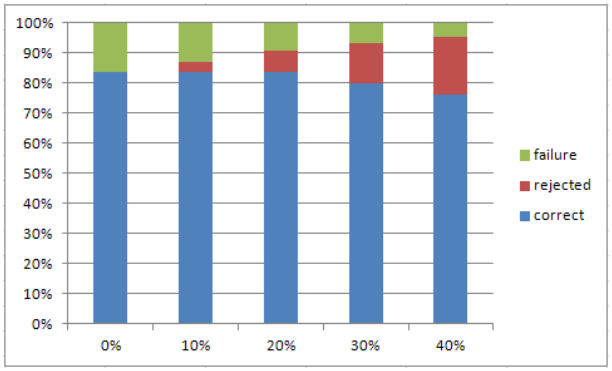
reference \ detection	srb	frb	orb	car	str	gra	veg	sha	pol	rej
srb	85,7	2,0	6,1	0,0	0,0	0,0	2,0	2,0	0,0	2,0
frb	0,0	95,2	4,8	0,0	0,0	0,0	0,0	0,0	0,0	0,0
orb	10,2	0,0	83,1	0,0	0,0	1,7	0,0	3,4	0,0	1,7
car	0,0	0,0	1,8	27,3	0,0	9,1	18,2	9,1	0,0	34,5
str	0,0	3,4	0,0	0,0	75,9	3,4	3,4	13,8	0,0	0,0
gra	0,0	2,3	0,0	0,0	2,3	62,8	23,3	2,3	2,3	4,7
veg	0,0	0,0	0,0	1,8	1,8	5,4	85,7	1,8	0,0	3,6
sha	0,0	0,0	0,0	0,0	0,0	5,4	18,9	56,8	0,0	18,9
pol	0,0	16,7	0,0	0,0	0,0	5,6	5,6	5,6	22,2	44,4

(b) The same dataset. This time we the neighbor-infomation node of the Bayes net was not used.

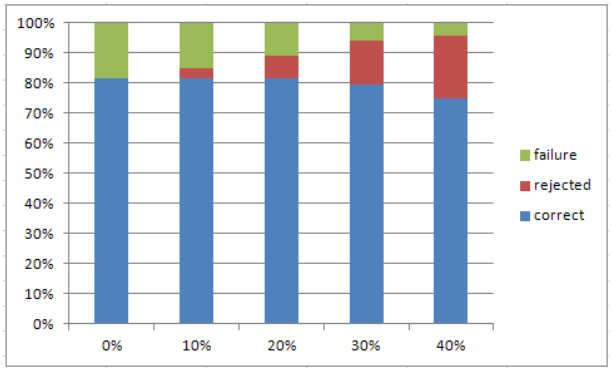
Figure 6.8: The figures show some experiments where the Bayes Net was modified to show the (missing) influence of some observations.



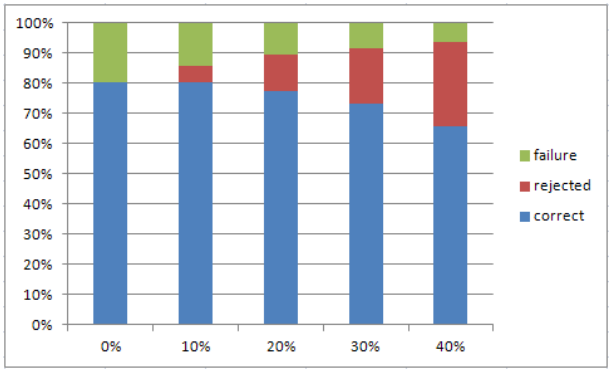
(a) Results of the region classification for the dataset GRA



(b) Results of the region classification for the dataset GRZ



(c) Results of the region classification for the dataset BOI



(d) Results of the region classification for the dataset TOY

Figure 6.9: The overall results for the four datasets with different acceptance thresholds. We can observe that increasing the threshold first reduces the misclassifications. When we increase further ($\geq 20\%$) also the weak classifications, that are accepted as correct, are rejected.

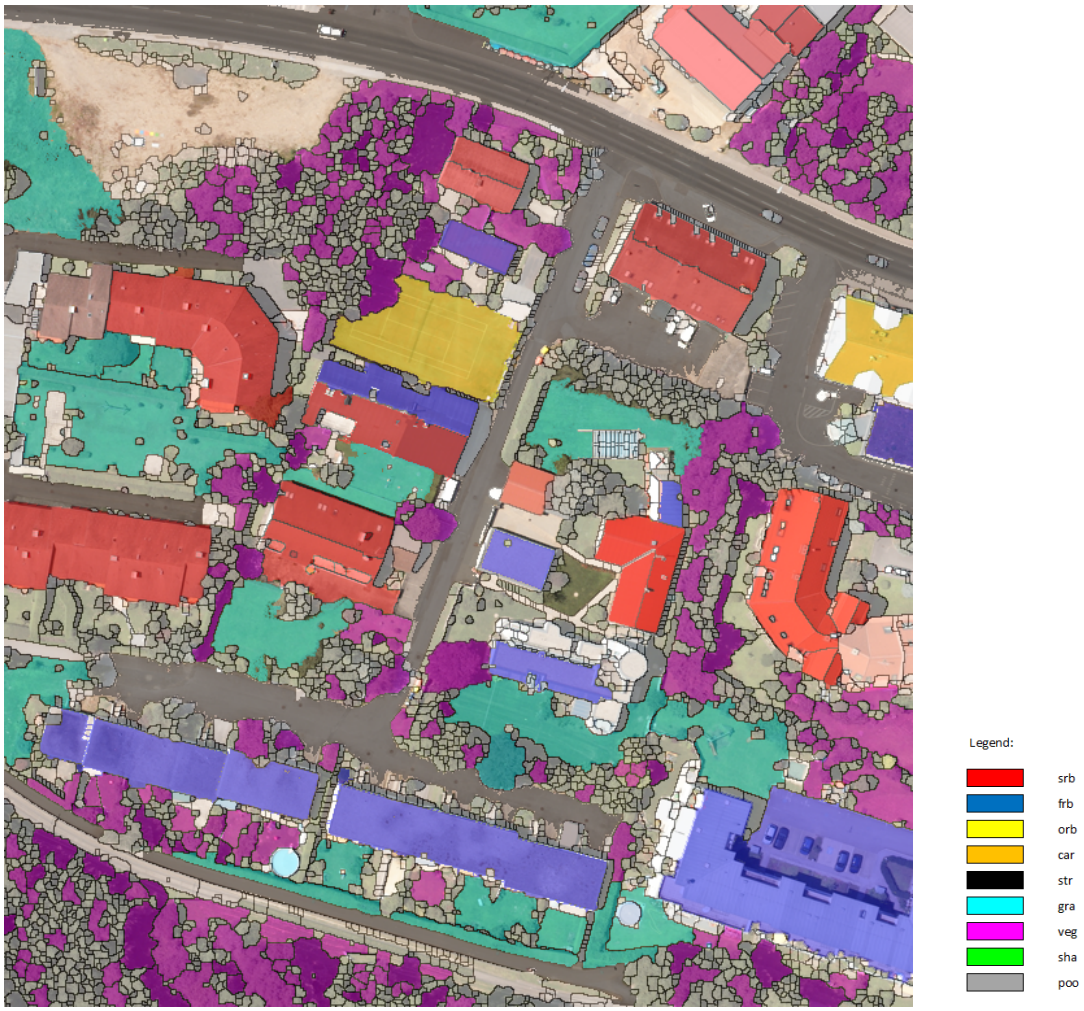


Figure 6.10: The object layer of the dataset Graz-Andritz. The image regions are colored according to the classification result of the corresponding object nodes.

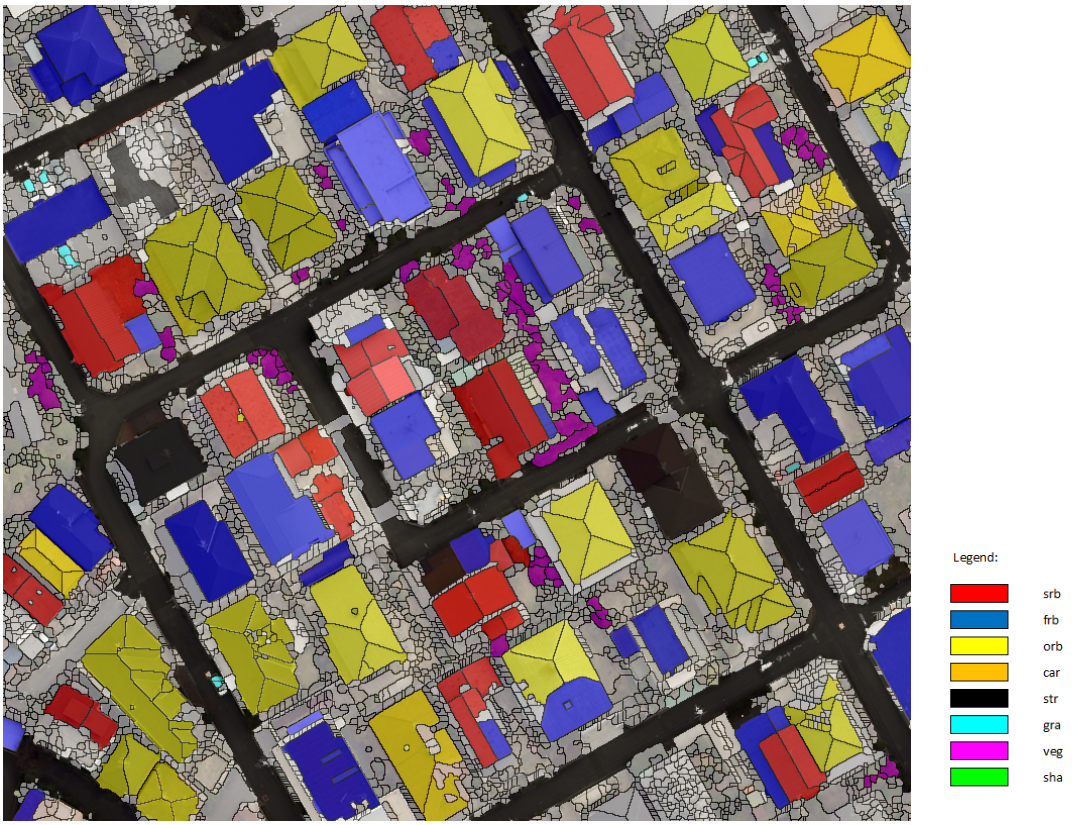


Figure 6.11: The object-layer of Toyonaka. The image regions are colored according to the classification result of the corresponding object nodes.

Applying the model to different data sets shows results of about 80% correct interpretations. Inside the classes for the roof detection the correct classifications are between 80% and 90%. The results are very stable and hold also for images from other data sets than the learned ground truth.

The results show, that the observed features contribute in different strength to the interpretation. Color for example contributes to the distinction between the classes vegetation and grass land. The geometrical observations are mainly the same for these type of classes, so the color information gains influence. The same can be examined for the neighbor information: Is is especially useful for object classes that are mixed with other objects, e.g. the multi colored rectangular cars.

The results show another important issue: The classification only with geometric features works very stable. The roof classes, especially the class *other roof building*, have a complicated structure. Even the Toyonaka dataset is interpreted here successfully. In many building detectors, the red color of the roof tile has a strong impact, and leads

(a) (b)

Figure 6.12: These two images were identified as different scene types than registered in the ground truth database. The left image is part of the BOI dataset and was labeled as urban scene. The right one is part of the GRA dataset and was labeled as rural.

due to the number of red roofs in middle Europe to a good prior. In our approach it is possible to interpret a scene correctly using the geometry of the objects.

Chapter 7

Conclusion and outlook

In this thesis we present an approach for the automatical interpretation of aerial images. We develop a statistical model that is able to classify and interpret image regions in a hierarchical way. With this model we can detect and localize multiple instances of predefined objects and create an interpretation of the complete scene. The results are stable against image distortions and deficiencies of the used feature extraction algorithms; the ontology for the interpretation is not implemented in a fixed way. It is tought to the algorithm in a learning step.

The model is a multi-level bayes net that interprets the content of a region adjacency graph. The bayes net is parametrized by the probability distributions and the net structure. These are acquired in a previous learning step. The novelty of this work is the learning of a Bayes net containing objects of variable structure. Threrefor we use a hierarchical region adjacency graph to efficiently code spatial information for the introduction into a bayes net. Using the region graph, we are able to introduce a variety of geometric, texture and topology observations. Because the bayes net is modeled as a tree structure on top of the region adjacency graph, we can propagate the evidence of the observations efficiently while restricting the search space for the spatial neighbor information. The information of the image regions are aggregated to cliques- and objects-nodes in the bayes net. The structure of the bayes net is not fixed a priori. It is learned together with the probability distributions of the nodes and creates a visual words vocabulary for the region and clique nodes of the bayes net.

The results for the interpretation of aerial images show an interpretation rate of approximately 80 percent correct classifications. Among the building classes the classification results are between 80 and 90 %. For building detection we reach results of over 90 percent. The classification is comparable to other classification algorithms. The

benefit here is, that we are able to combine the parameters of different image domains and to use these for a correct interpretation.

The approach provides a fast interpretation of the scene, as well in the learning task as in the interpretation. Although we tested the model for aerial image interpretation, it will work for any other scene that complies with the underlying image model. For example the detection of complex objects in industrial applications or categorization of image databases could be possible.

The challenge for the future is to adapt this approach to systems like e.g. google Earth or other image databases containing aerial or satellite data. The used algorithms are able to work in parallel, so the processing of huge amount of data in the "'cloud"' is very efficient. Using this interpretation, there are phenomenona tracable like the long term change in housing and settlements in the delevoping countries or the desertification e.g. in Africa. One scale smaller it is possible to track changes for an automatic update of maps.

Technically, it is interesting to expand the approach in several ways. One direction would be to include knowledge of the third dimension of the scene. Terrestrial images have more depth information and occlusion than aerial images. Therefore other features would be helpful. Also in the aerial images some information about the third dimension would be helpful in a next step, e.g. the direction of shadows and the distortion due to height differences.

Another goal is to widen and differentiate the hierarchical ontology of the data. Therefore we would need a more detailed groundtruth database and a consistent ontology for working on defined subsets of scenes.

The detection results could also be improved by using additional sensor data. By adding other image channels like near infrared or even 3-D information by LIDAR or SAR measurements, the evidence of the objects in the image scene can be inferred more easily.

List of Figures

List of Tables

List of Algorithms

Bibliography

Agarwal, Ankur and Bill Triggs [2006]. "Hyperfeatures – Multilevel Local Coding for Visual Recognition". In: LNCS 3951. Ed. by Aleš Leonardis et al., pp. 30–43.

Agarwal, Shivani and Dan Roth [2002]. "Learning a Sparse Representation for Object Detection." In: *ECCV*. Vol. 4, pp. 113–130.

Almuallim, H. and T. G. Dietterich [1991]. "Learning with many irrelevant features". In: *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91)*. Vol. 2. Anaheim, California: AAAI Press, pp. 547–552. URL: `citeseer.ist.psu.edu/almuallim91learning.html`.

Aly, Mohamed et al. [2011]. *Using More Visual Words in Bag of Words Large Scale Image Search*. Tech. rep. Caltech, USA.

Amores, J. et al. [2005]. "Fast spatial pattern discovery integrating boosting with constellations of contextual discriptors." In: *CVPR*. Vol. 2, pp. 769–774.

Bangham, J. Andrew et al. [1999]. "Scale-space Trees and Applications as Filters for Stereo Vision and Image Retrieval". In: *BMVC*, pp. 113–143.

Belongie, Serge et al. [2002]. "Shape Matching and Object Recognition Using Shape Contexts." In: *IEEE Trans. Pattern Anal. Mach. Intell.* 24.4, pp. 509–522.

Berg, A. et al. [2005]. "Shape matching and object recognition using low distortion correspondence". In: *CVPR*. Vol. 1, pp. 26–33.

Biederman, I. [1987]. "Recognition-by-Components: A Theory of Human Image Understanding". In: *Psychological Review* 94, pp. 115–147.

Bishop, Christopher M. [2006]. *Pattern Recognition and Machine Learning*. Springer Verlag.

Boykov, Yuri Y and M-P Jolly [2001]. "Interactive graph cuts for optimal boundary & region segmentation of objects in ND images". In: *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*. Vol. 1. IEEE, pp. 105–112.

Braun, Carola [1994]. "Interpretation von Einzelbildern zur Gebäuderekontruktion". PhD thesis. Institut für Photogrammetrie Bonn.

Braun, Carola et al. [1994]. "Models for Photogrammetric Building Reconstruction". In: *Computer & Graphics* 19.1, pp. 109–118.

Broadway, Jonathan Yedidia et al. [2000]. "Generalized Belief Propagation". In: pp. 689–695.

Brunn, A. and W. Förstner [1995]. "Model-based 2D-Shape Recovery". In: *DAGM*.

Buntine, Wray L. [1994]. "Operations for Learning with Graphical Models". In: *Journal of Artificial Intelligence Research* 2, pp. 159–225.

Burl, Michael C. et al. [1998]. "A Probabilistic Approach to Object Recognition Using Local Photometry and Global Geometry." In: *ECCV (2)*, pp. 628–641.

Cao, Liangliang and Li Fei-Fei [2007]. "Spatially coherent latent topic model for concurrent object segmentation and classification". In: *IEEE Intern. Conf. in Computer Vision (ICCV)*.

Carneiro, Gustavo and David Lowe [2006]. "Sparse Flexible Models of Local Features". In: LNCS 3953. Ed. by Aleš Leonardis et al., pp. 29–43.

Chang, Peng and John Krumm [1999]. "Object Recognition with Color Cooccurrence Histograms". In: *CVPR*.

Clarenz, Ulrich et al. [2004]. "On level set formulations for anisotropic mean curvature flow and surface diffusion". In: International Series of Numerical Mathematics 149. Ed. by Axel Voigt, pp. 227–238.

Collins, R. et al. [1998]. "The ASCENDER System Automated Site Modeling from Multiple Aerial Images". In:

Crandall, D. et al. [2005]. "Spatial priors for part-based recognition using statistical models". In: *CVPR*. Vol. 1, pp. 10–17.

Crowley, James L. et al. [2003]. *Fast Computation of Characteristic Scale Using a Half-Octave Pyramid*.

Dempster, A. et al. [1976]. "Maximum likelihood from incomplete data via the EM algorithm". In: *JRSS* 39, pp. 1–38.

Dickinson, Sven et al. [2005]. "Object Categorization and the Need for Many-to-Many Matching". In: *DAGM*.

Dissard, O. et al. [1997]. "Above-Ground Objects in Urban Scenes from Medium Scale Aerial Imagery". In: *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*. Ed. by A. Gruen et al. Birkhäuser, Basel, pp. 183–192.

Dorkó, Gy. and C. Schmid [2003]. "Selection of Scale-Invariant Parts for Object Class Recognition". In: *ICCV*.

Douglas, D.H. and T.K. Peucker [1973]. "Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or Its Caricature". In: *The Canadian Cartographer* 10.2, pp. 112–122.

Drauschke, Martin [2011]. "Ein hierarchischer Ansatz zur Interpretation von Gebäudeaufnahmen". PhD thesis. Institute of Photogrammetry, University of Bonn.

Drauschke, Martin et al. [Sept. 2006]. "Stabilität von Regionen im Skalenraum". In: 15. Ed. by Eckhardt Seyfert, pp. 29–36.

Elkan, Charles [1997]. *Boosting and naive bayes learning*. Tech. rep. University of California, San Diego.

Farhadi, Ali et al. [Dec. 31, 2009]. "Describing objects by their attributes." In: *CVPR*. IEEE, pp. 1778–1785. ISBN: 978-1-4244-3992-8. URL: `http://dblp.uni-trier.de/db/conf/cvpr/cvpr2009.html#FarhadiEHF09`.

Fei-Fei, L. et al. [2003]. "A Bayesian Approach to Unsupervised One-Shot Learning of Object Categories." In: *Proceedings of the 9th International Conference on Computer Vision, Nice, France*, pp. 1134–1141.

— [2004]. "A Bayesian approach to unsupervised one-shot learning of object categories". In: *ICCV*.

Felzenszwalb, Pedro F. and Daniel P. Huttenlocher [2005]. "Pictorial Structures for Object Recognition." In: *International Journal of Computer Vision* 61.1, pp. 55–79.

Fergus, R. et al. [2005]. "A sparse object category model for efficient learning and exhaustive recognition". In: *CVPR*. Vol. 1, pp. 380–387.

Ferrari, Vittorio et al. [2006]. "Object Detection by Contour Segment Networks". In: LNCS 3953. Ed. by Aleš Leonardis et al., pp. 14–28.

Fischer, André et al. [1998]. "Extracting Buildings from Aerial Images Using Hierarchical Aggregation in 2D and 3D". In: *Computer Vision and Image Understanding: CVIU* 72.2, pp. 185–203. URL: `citeseer.ist.psu.edu/fischer98extracting.html`.

Fischer, A. et al. [1999]. "On the Use of Geometric and Semantic Models for Component–Based Building Reconstruction". In: *SMATI 99*, pp. 101–120.

Fischler, M. A. and R. A. Elschlager [1973]. "The representation und matching of pictorial structures". In: *IEEE Trans. Computers*, pp. 67–92.

Förstner, W. [1994]. "A Framework for Low Level Feature Extraction". In: *European Conference on Computer Vision*, pp. 383–394.

Förstner, Wolfgang, ed. [2009]. *Vorlesung Photogrammetrie*. Nussallee 17, 53121 Bonn: Universität Bonn.

Friedman, Nir [1997]. "Learning belief networks in the presence of missing values and hidden variables". In: *Proc. 14th International Conference on Machine Learning*. Morgan Kaufmann, pp. 125–133. URL: citeseer.ist.psu.edu/friedman97learning.html.

— [1998]. "The Bayesian Structural EM Algorithm". In: *UAI*, pp. 129–138. URL: citeseer.ist.psu.edu/article/friedman98bayesian.html.

Friedman, Nir et al. [1997]. "Bayesian Network Classifiers". In: *Machine Learning* 29.2-3, pp. 131–163. URL: citeseer.ist.psu.edu/friedman97bayesian.html.

Fuchs, C. [1998]. "Parameterarme Verfahren zur Extraktion polymorpher Bildstrukturen und ihre topologische und geometrische Gruppierung für die Bildsegmentierung". PhD thesis. Deutsche Geodätische Kommission, München.

Green, P. [1995]. *Reversible jump Markov chain Monte Carlo computation and Bayesian model determination*. URL: citeseer.ist.psu.edu/green95reversible.html.

Harvey, Richard et al. [1997]. "Scale-Space Filters and Their Robustness". In: *Scale-Space*, pp. 341–344.

Heckerman, David et al. [1995]. "Learning Bayesian Networks: The Combination of Knowledge and Statistical Data." In: *Machine Learning* 20.3, pp. 197–243.

Helmer, S. and D. Lowe [2004]. "Object recognition with many local features". In: *Workshop on Generative Model Based Vision*.

Ihler, Alexander T. et al. [2005]. "Loopy belief propagation: Convergence and effects of message errors". In: vol. 6, pp. 905–936.

Jähne, B. [1989]. *Digitale Bildverarbeitung*. Springer–Verlag.

Jaynes, C. et al. [1997]. "Building Reconstruction from Optical and Range Images". In: *Workshop on Semantic Modeling for the Acquisition of Topographic Information from Images and Maps, SMATI '97*. Ed. by W. Förstner and L. Plümer. To appear.

Jegou, Herve et al. [Sept. 2012]. "Aggregating Local Image Descriptors into Compact Codes". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 34.9, pp. 1704–1716. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2011.235. URL: http://dx.doi.org/10.1109/TPAMI.2011.235.

Jordan, Michael I. et al. [1999]. "An Introduction to Variational Methods for Graphical Models". In: *Machine Learning* 37.2, pp. 183–233.

Kadir, Timor and Michael Brady [Nov. 2001]. "Saliency, Scale and Image Description". In: *International Journal of Computer Vision* 45.2, pp. 83–105.

Klonowski, J. and K.R. Koch [1997]. "Two Level Image Interpretation Based on Markov Random Fields". In: *Semantic Modeling for the Aquisition of Topographic Information from Images and Maps*. Ed. by Wolfgang Förstner and Lutz Plümer, pp. 37–58.

Koller, Daphne and Nir Friedman [2009]. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press. ISBN: 0262013193, 9780262013192.

Korc, Phili [2012]. "Tractable Learning for a Class of Global Discriminative Models for Context Sensitive Image Interpretation". PhD thesis. Department of Photogrammetry, University of Bonn.

Kulschewski, Kai [1997]. "Building Recognition with Bayesian Networks". In: *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps: SMATI97*. Ed. by Wolfgang Förstner and L. Plümer. Basel, Switzerland: Birkhäuser Verlag.

— [1999]. "Modellierung von Unsicherheiten in dynamischen Bayes-Netzen zur qualitativen Gebäudeerkennung". PhD thesis. Universität Bonn.

Kumar, M. P. et al. [2005]. "OBJ CUT". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego*.

Kumar, Sanjiv and Martial Hebert [2003]. "Man-Made Structure Detection in Natural Images using a Causal Multiscale Random Field." In: *CVPR (1)*, pp. 119–126.

Landes NRW, Staatskanzlei des [Aug. 2011]. URL: `http://www.nrw.de/landesregierung/kampfmittelraeumdienst-entschaerft-230-grosse-bomben-11430/`.

Lauritzen, Steffen L. [1996]. *Graphical Models*. Oxford Science Publications.

Leibe, Bastian and Bernt Schiele [2004]. "Scale-Invariant Object Categorization Using a Scale-Adaptive Mean-Shift Search." In: *DAGM-Symposium*, pp. 145–153.

Leibe, B. et al. [May 2004]. "Combined object categorization and segmentation with an implicit shape model". In: *Proceedings of the Workshop on Statistical Learning in Computer Vision*. Prague, Czech Republic.

Li, Li-Jia and Li Fei-Fei [2007]. "What, where and who? Classifying events by scene and object recognition". In: *IEEE 11th International Conference on Computer Vision*, pp. 1–8.

Li, Li-jia et al. [2010]. "Object Bank: A High-Level Image Representation for Scene Classification & Semantic Feature Sparsification". In: *Advances in Neural Information Processing Systems 23*. Ed. by J.D. Lafferty et al. Curran Associates, Inc., pp. 1378–1386. URL: `http://papers.nips.cc/paper/4008-object-bank-a-high-level-image-representation-for-scene-classification-semantic-feature-sparsification.pdf`.

Li, Yin et al. [2005]. "Object Class Recognition using Images of Abstract Regions". In: *ICPR*.

Lin, C. and R. Nevatia [1995]. "3-D Descriptions of Buildings from an Oblique View Aerial Image". In: *IEEE International Symposium on Computer Vision*, pp. 377–382.

— [1996]. "Buildings Detection and Description from Monocular Aerial Images". In: *ARPA Image Understanding Workshop*.

Lin, Chungan and Ramakant Nevatia [1998]. "Building Detection and Description from a Single Intensity Image". In: *Computer Vision and Image Understanding* 72.2, pp. 101–121.

Lindeberg, Tony [1990]. "Scale-Space for Discrete Signals." In: *IEEE Trans. Pattern Anal. Mach. Intell.* 12.3, pp. 234–254.

— [1996]. "Scale-space theory: A framework for handling image structures at multiple scales". In: *CERN School of Computing*.

Lowe, David G. [2004]. "Distinctive Image Features from Scale-Invariant Keypoints". In: *International Journal on Computer Vision* 60.2, pp. 91–110. ISSN: 0920-5691.

Loy, Gareth and Jan-Olof Eklundh [2006]. "Detecting Symmetry and Symmetric Constellations of Features". In: LNCS 3952. Ed. by Aleš Leonardis et al., pp. 508–521.

Matas, Jiri et al. [2002]. "Robust Wide Baseline Stereo from Maximally Stable Extremal Regions". In: *BMVC*. Vol. 1, pp. 384–393.

Mayer, Helmut [2000]. "Scale-Space Events for the Generalization of 3D-Building Data". In: *International Archives of Photogrammetry and Remote Sensing*. Vol. 33, pp. 639–646.

Meidow, Jochen [2000]. "Gemeinsame Segmentierung und Interpretation digitaler Luftbilder mit Hilfe der Bayes-Statistik". PhD thesis. Institut für theoretische Geodäsie Uni Bonn.

Mikolajczyk, Krystian and Cordelia Schmid [2003]. "A performance evaluation of local descriptors." In: *CVPR (2)*, pp. 257–263.

Murphy, Kevin P. [2001]. "Learning Bayes net Structure from sparse data sets". In:

Murphy, Kevin P [2012]. *Machine learning: a probabilistic perspective*. Cambridge, MA.

Murphy, Kevin, Antonio Torralba, Daniel Eaton, et al. [2005]. "Object detection and localization using local and global features". In: *Sicily workshop on object recognition*.

Murphy, Kevin, Antonio Torralba, and William Freeman [2003]. "Using the Forest to See the Trees:A Graphical Model Relating Features, Objects and Scenes". In: *NIPS'03 (Neural Info. Processing Systems)*.

Nguatem, William et al. [2013]. "Roof Reconstruction from Point Clouds Using Importance Sampling". In: *City Models, Roads and Traffic 2013 (CMRT13)*. Ed. by Franz Rottensteiner et al. Vol. II. 3. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, pp. 73–78.

Niebles, Juan Carlos and Li Fei-Fei [2007]. "A Hierarchical Model of Shape and Appearance for Human Action Classification". In:

Noronha, Sanjay and Ramakant Nevatia [1997]. "Detection and Description of Buildings from Multiple Aerial Images." In: *CVPR*, pp. 588–594.

— [2001]. "Detection and Modeling of Buildings from Multiple Aerial Images." In: *IEEE Trans. Pattern Anal. Mach. Intell.* 23.5, pp. 501–518.

Oliva, Aude and Antonio Torralba [May 2001]. "Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope". In: *International Journal of Computer Vision* 42.3, pp. 145–175.

Oliva, Aude, Antonio B. Torralba, et al. [2003]. "Top-down control of visual attention in object detection." In: pp. 253–256.

Paparoditis, N. et al. [1998]. "Building Detection And Reconstruction From Mid- And High-Resolution Aerial Imagery". In: *Computer Vision and Image Understanding* 72.2, pp. 122–142.

Pearl, Judea [1988]. *Causality ???* 1st. Cambridge University Press.

— [2000]. *Causality.* 2nd. Cambridge University Press.

Philbin, J. et al. [2007]. "Object Retrieval with Large Vocabularies and Fast Spatial Matching". In: *IEEE Conference on Computer Vision and Pattern Recognition.*

Pradhan, Malcolm and Paul Dagum [1996]. "Optimal Monte Carlo Estimation of Belief Network Inference". In: pp. 446–453.

Rabinowich, Andrew et al. [2007]. "Objects in Context". In: *ICCV.*

Richardson, Sylvia and Peter J. Green [1997]. "On Bayesian analysis of mixtures with an unknown number of components". In: *Institute of International Economics Project on International Competition Policy," COM/DAFFE/CLP/TD(94)42.*

Ripley, Brian D [2007]. *Pattern recognition and neural networks.* Cambridge University Press.

Rodrigues, Paulo Sérgio and Arnaldo de Albuquerque Araújo [2002]. "A Region-Based Object Recognition Algorithm." In: *SIBGRAPI*, pp. 283–.

Rosenhahn, Bodo et al. [2006]. "A Comparison of Shape Matching Methods for Contour Based Pose Estimation". In: *Lecture Notes in Computer Science* 4040, pp. 263–276. DOI: 10.1007/11774938_21. URL: http://www.springerlink.com/content/157u4w35l1m21316/.

Rother, Carsten et al. [2004]. ""GrabCut – interactive foreground extraction using iterated graph cuts". In: *ACM TRANS. GRAPH*, pp. 309–314.

Rottensteiner, Franz [Nov. 2003]. "Automatic Generation of High-Quality Building Models from Lidar Data". In: *IEEE Computer Graphics and Applications* 23.6, pp. 42–50.

Russakovsky, Olga et al. [2013]. "Detecting avocados to zucchinis: what have we done, and where are we going?" In: *International Conference on Computer Vision (ICCV)*.

Russell, B. C. et al. [2005]. "LabelMe: a database and web-based tool for image annotation". In: *MIT AI Lab Memo AIM-2005-025*.

Schröder-Brzosniowsky, Michael [1999]. "Stochastic Modeling of Image Content in Remote Sensing Image Analysis". PhD thesis. ETH Zürich.

Torralba, Antonio B. [2003]. "Contextual Priming for Object Detection." In: *International Journal of Computer Vision* 53.2, pp. 169–191.

Torralba, Antonio B. et al. [2003]. "Context-based vision system for place and object recognition." In: *ICCV*, pp. 273–280.

Torralba, Antonio et al. [2004]. "Sharing Features: Efficient Boosting Procedures for Multiclass Object Detection". In: pp. 762–769.

Tu, Peter et al. [1999]. "Recognizing Objects Using Color-Annotated Adjacency Graphs". In: *Shape, Contour and Grouping in Computer Vision*. London, UK: Springer-Verlag, pp. 246–263. ISBN: 3-540-66722-9.

Valiant, Leslie [2013]. *Probably Approximately Correct*. Basic Books.

Vogel, Julia [2004]. "Semantic Scene Modeling and Retrieval". PhD thesis. ETH Zürich.

Vogel, Julia and Bernt Schiele [Sept. 2004]. "A Semantic Typicality Measure for Natural Scene Categorization". In: *Pattern Recognition Symposium DAGM'04*. Tübingen, Germany.

Weber, Markus et al. [2000a]. "Towards Automatic Discovery of Object Categories." In: *CVPR*, pp. 2101–.

— [2000b]. "Unsupervised Learning of Models for Recognition." In: *ECCV (1)*, pp. 18–32.

Winn, John et al. [2005]. "Variational message passing". In: *Journal of Machine Learning Research* 6, pp. 661–694.

Yang, Micheal [2011]. "Hierarchical and Spatial Structures for Interpreting Images of Man-made Scenes Using Graphical Models". PhD thesis. Institute of Photogrammetry, University of Bonn.

Yps et al. [May 1976]. "Die Kamera, die echte Fotos macht". In: *Yps-Heft* 33, pp. 1–4.

Zhang, Wei et al. [2005]. "Object Class Recognition Using Multiple Layer Boosting with Heterogeneous Features." In: *CVPR (2)*, pp. 323–330.

Zia, M.Z. et al. [2013]. "Towards Scene Understanding with Detailed 3D Object Representations". In: *IEEE CVPR Workshop on Scene Understanding*.

Zoubin Ghahramani, Zoubin [1997]. "Learning Dynamic Bayesian Networks". In: *Lecture Notes in Artificial Intelligence*.