EuroSDR

Automated Extraction of Topographic Map Data from Remotely Sensed Imagery by Classification and Cartographic Enhancement

An Introduction to New Mapping Tools

Joachim Höhle, Bharath Bhushan Damodaran

EUROPEAN SPATIAL DATA RESEARCH

PRESIDENT 2023-25:
     Wolfgang Gold, Austria

VICE-PRESIDENT 2021 – 2023:
     Fabio Remondino, Italy

SECRETARY – GENERAL 2019 – 2023:
     Joep Crompvoets, Belgium

DELEGATES BY MEMBER COUNTRY:
     Austria: Wolfgang Gold, Gottfried Mandlburger
     Belgium: Eric Bayers
     Croatia: to be confirmed
     Cyprus: Elikkos Elia, Georgia Papathoma, Andreas Hadjiraftis,
     Denmark: Katerina Ritz
     Estonia: Tambet Tiits, Artu Ellmann, Evelyn Uuemaa
     Finland: Juha Hyyppä, Juha Kareinen
     France: Bénédicte Bucher, Aurélien Plyer
     Germany: Michael Hovenbitzer, Norbert Haala
     Greece: Lolonis Panagiotis
     Ireland: Paul Kane, Audrey Martin
     Norway: Jon Arne Trollvik, Ivar Maalen-Johansen
     Poland: Anna Bober, Krzysztof Bakuła
     Portugal: Mário Caetano, Rui Reis, Marisa Silva
     Slovenia: Dalibor Radovan, Peter Prešeren, Marjan Čeh
     Spain: Julián Delgado Hernández
     Sweden: Tobias Lindholm, Anders Rydén, Heather Reese
     Switzerland: André Streilein, Adrien Gressin
     The Netherlands: Martijn Rijsdijk, George Vosselman
     United Kingdom: David Henderson, Claire Ellul

ASSOCIATE MEMBERS AND THEIR REPRESENTATIVES:
     Digitaal Vlaanderen: Jo Van Valckenborgh
     Esri: Nick Land, Konrad Wenzel (nFrames)
     Field Group: Leif Erik Blankenberg
     Hexagon: Simon Musaeus
     ICGC: Julià Talaya
     IGI: Philipp Grimm
     RIEGL: Peter Rieger
     Vexcel: Michael Gruber
     1Spatial: Dan Warner

The official publications of EuroSDR are peer-reviewed.

Joachim Höhle, Bharath Bhushan Damodaran

"Automated Extraction of Topographic Map Data from Remotely Sensed Imagery
by Classification and Cartographic Enhancement"

Table of contents

AUTOMATED EXTRACTION OF TOPOGRAPHIC MAP DATA FROM REMOTELY SENSED IMAGERY BY CLASSIFICATION AND CARTOGRAPHIC ENHANCEMENT

An Introduction to New Mapping Tools

With 37 figures and 5 tables

Joachim Höhle [a], Bharath Bhushan Damodaran [b]

[a] Aalborg University

Aalborg, Denmark

[b] IRISA-OBELIX, Université de Bretagne-Sud

Vannes, France

(present address: InterDigital Inc., Rennes, France)

Index of Figures

Index of Tables

Preface

Courses about new developments in the mapping sciences have been carried out by EuroSDR since 2002. They were designed as blended learning, which means that teachers and participants/students met face-to-face in a two-day pre-course seminar and the e-learning took place by means of virtual communication for two weeks thereafter. In this period the teachers were present on the Internet answering questions and correcting the assignments within a short time.

The course "Automatic topographic mapping through description and classification of remotely sensed imagery and cartographic enhancement" was part of the EuroSDR Educational Service in 2018 and 2019 which were held together with professor Sébastien Lefèvre, Université Bretagne Sud. Now, the course material is opened for self-learning, but no support is given anymore by the teachers. However, the course material has been adapted by the authors of the book to this new condition. This book describes the course material of this e-learning course, and it can be used as supplementary material to the course content that is available online. It may also serve as an example for designing course material of new courses.

INTRODUCTION

Topographic map data are a prerequisite for economic development in all parts of the world. It is the task of governmental and private mapping organizations to generate and maintain such data. Efficient and accurate methods are required. Universities and research groups support the efforts to develop new methods. Also, motivated citizens want to help in the updating of existing map data. Such 'crowd sourcing' is very much in demand and should be encouraged by providing education and tools.

It is the first objective of this book to introduce new technologies for map production and map updating. The big changes in technology in mapping techniques and the demands for complete and up-to-date topographic data bases also require education. Therefore, e-learning courses have been carried out by EuroSDRs educational service. The education on the Internet has been supported by a pre-course seminar and a support by the teachers on the Internet. Two courses on the topic "Automatic topographic mapping through description and classification of remotely sensed imagery and cartographic enhancement" have been carried out. It has also been an objective to ease self-studies and apply 'Learning by doing'. The well-documented software tools will be used in the 'assignments' the participants are asked to do. This 'hands-on' approach helps to understand the applied tools and inspires to extension and adaptation to special requirements at mapping organizations. The book will also serve as an example for courses of other topics and help future teachers to design course material to be used in self-learning. Pedagogical principles take precedence of writing for academic journals.

To extract topographic map data automatically from remotely sensed data is a sophisticated process. The source data is imagery of aerial cameras, satellite sensors, and point clouds of airborne laser scanning. Data of existing data bases, e.g., digital elevation models (DEM), may also be at disposal. In general, mapping of new as well as removal of obsolete topographic objects has two tasks:

- recognition and interpretation of changes and
- registration and cartographic enhancement of the detected objects.

These tasks may be solved in an automatic process by means of:

- generation of a land cover map (LCM) of several classes using DSM-based orthoimages and
- extraction of topographic objects from the LCM and their cartographic enhancement.

The first task is carried out by a supervised classification. Advanced methods have been developed in the last few years enabling accurate and fast results. These methods are based on 'machine learning' techniques. Some information about the topographic objects must be known in advance. These so-called features must characterize the objects and enable separation from other objects (classes). The automatic generation of proper features is important for a successful classification. In the provided software tools a variety of features are derived, e.g., vegetation index, height above ground, and profiles of available spectral bands. The second task, extraction of topographic objects from the generated land over map in raster format and the conversion to vector data of high geometric and cartographic quality is solved by a package of several programs. This task is difficult for automatic solutions due to the diversity of topographic objects. The applied solution enables a graphical output of the intermediate steps together with the possibility for interactions to achieve accurate and complete results. The software tools are well-documented, which helps to understand the solution and to adapt to special requirements.

The EduServ courses of EuroSDR use the learning management system "Moodle". This system has various possibilities to present a topic and to carry out the learning. The components for the knowledge transfer are this book, several blogs and articles, videos, and software tools. For the self-evaluation there are quizzes and assignments including answers and solutions. An important part of

the learning are the assignments which the participants are asked to carry out. Working with the software tools is suggested and collaborative learning is supported. It is the conviction of the teachers that the practical use of software tools contributes very much to good understanding of the topic. The general process of learning within a course is depicted in Figure 1.
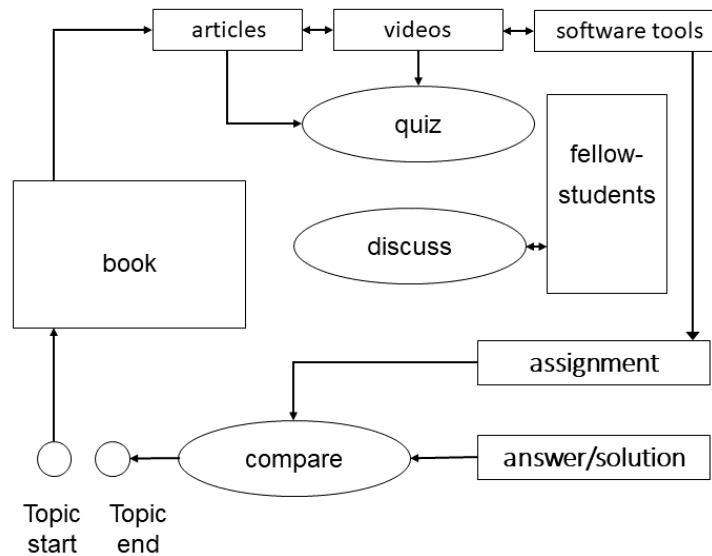
Figure 1: Strategy in the learning (modified after Höhle, 2008).

In the learning management system 'Moodle', the course material is structured in Modules and one course has usually 4 to 6 Modules. The activities of the participants are reading of the literature, looking to videos, solving quizzes, discussion with the fellow students, and comparing their work for quiz and assignments with the given answers/solutions. These components are explained in the following.

Book

Each chapter is introduced by a short text informing about the content of the chapter and the prepared learning aids. The theoretical background together with the applied formulas are presented. References are added at the end of each chapter.

Blogs and articles

The learning is started by reading short blogs and longer articles. They are either on-line in Moodle or may be accessed by using the provided link to an Internet address.

Videos

The provided videos may be recordings of lectures of conferences or slide shows with explanations as text or speech.

Quiz

Multiple-choice tests are presented to the learner. The answers are immediately checked by the system. The Moodle system will record the number of attempts. The correct answer/solution is explained by means of texts and graphs.

Assignments

The participants of the course are asked to work with assignments. They may wish to consult the answers/solutions which are available in the learning system as texts and graphs.

Software tools

The software tools realizing single steps of classification and enhancement are provided. The programs are well-documented. Together with provided data sets, the display of intermediate results by text and graphs may be experienced. Installation of the programs may become necessary, but instructions how to do it are part of the learning material. The content of the programs is explained.

By means of the provided answers to quizzes and assignments a self-evaluation is possible and a progress in the learning may be experienced. We believe that an active work with the course material will help to understand the topics of the course. Also, modification and extension of the software tools is possible. The learning material of the course "Automatic topographic mapping through description and classification of remotely sensed imagery and cartographic enhancement" is described in context, in detail, and in a homogeneous mode.

This book will give an example of course material adapted to self-learning. The importance of the software tools is weighted high, and the previous material is extended in this respect. All provided software is open source. The teachers encourage the participants to own developments and modifications.

This book has the following chapters. The principles of image analysis for topographic mapping are explained in *Chapter 1*. The extraction of spatial-spectral features and the possible data sources are discussed as well. Various classification methods based on machine learning are dealt with in *Chapter 2*. The resulting land cover maps are assessed and then enhanced to vector data. The applied methodologies are explained in *Chapter 3*. Processed examples comprising buildings, roads, and trees are presented in *Chapter 4*. *References* are added to each chapter. The *Appendix* includes the learning material in the learning system "Moodle" (literature to read, quizzes and assignments including the solutions and answers, description of the applied software tools and data).

References

Höhle, J., 2008. Experiences with combining e-learning courses with face-to-face education. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXVII, part B6a, commission VI, ISSN 1682-1750, pp. 59-64.

CHAPTER 1. Principles of image analysis for topographic mapping

This chapter introduces the basic concepts of image analysis and the challenges involved in automatic mapping from high-resolution images. Furthermore, the different types of data sources that can successfully be applied in automated mapping of topographic objects are also covered.

1.1 Basic concepts of image analysis

Methods from image analysis will be used to detect and extract topographic objects in the applied imagery. To map them in correct position, the imagery must be rectified. Furthermore, the mapped position should be independent from the object's elevation. Such imagery is the DSM-based ortho-image which may be used in the process of classification. Identification and mapping are then realized in one step. The classification of images requires some attributes which must describe the topographic objects and separate them from other objects. There are different types of attributes, basic and spatial attributes. The first group is based on the position of one pixel, the second group also uses the information of the neighbourhood. A third group are attribute profiles (AP). For each pixel in the image, a feature vector is established containing all selected attributes. The classification method uses the feature vector to predict the class (object) of each pixel.

1) Basic attributes

Basic attributes are the intensities of the available spectral bands. At aerial cameras, these are the intensities (I) of the basic colours (**R**ed, **G**reen, **B**lue) and of the near Infrared (**NIR**) channel. Other basic attributes are the elevations of the terrain surface and of the terrain ground. From these data other basic attributes may be derived. These are, for example, the normalized difference vegetation index ($NDVI$) and the height above ground ($nDSM$). They are derived by equations (1) and (2).

$$NDVI = (I_{NIR} - I_R) / (I_{NIR} + I_R) \tag{1}$$

$$nDSM = DSM - DTM \tag{2}$$

where

I_{NIR} = intensity of the NIR-band, I_R = intensity of the R-band, DSM = digital surface model, DTM = digital terrain model.

The $NDVI$ is used to detect vegetated areas and the $nDSM$ to detect buildings and other elevated objects.

2) Spatial attributes

Spatial attributes also include pixels or cells of the neighbourhood. In our applications, we use the standard deviation of the intensities or elevations in a 5x5 pixels large neighbourhood of one pixel (sigma I$_{5x5}$, sigma Z$_{5x5}$). Other spatial attributes are based on the co-occurrence matrix, e.g., the Haralick features which characterize the texture of an object.

3) Attribute profiles

An attribute profile (AP) consists of several filtered images which are generated from the bands of the applied image. For each of the bands a tree of connected components (CC) is created and then pruned using a threshold (T) for an attribute of the CC at different intensity levels. For example, a *'maxTree'* -

algorithm can be applied where the filtering of the created CC is carried out using the attribute *"area"* for thresholding. The pruned pixels are re-allocated to the CC of the node with a lower level of intensity. Such filtering of the image merges small bright regions with larger darker regions. This procedure is named thinning attribute filtering (Dalla Mura et al., 2010; Damodaran et al., 2017; Pham et al., 2018). Figure 2 depicts the principle of this filtering.
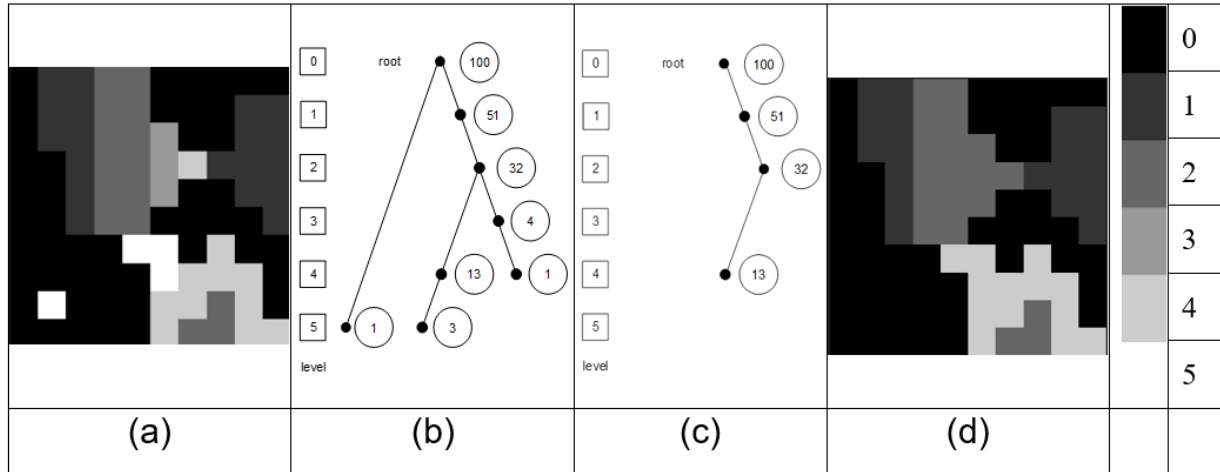
Figure 2: Principle of thinning attribute filtering. Original image
(a), maxTree with area size and levels (b), pruned tree after applying the threshold $T_{area} > 5$ pixels
(c), and attribute (*"area"*-) filtered image (d). (Source: Höhle, 2021).

This filtering removes small bright areas within dark areas, e.g., at the roofs of buildings, and thus enables a better detection of buildings. Also, a thickening attribute filtering could be created when using the *'minTree'* – algorithm. Applying other thresholds (T_{area}), will lead to different filtering. With a set of thresholds, a set of filtered images for one spectral band may be generated. Also, differential attribute profiles (DAP) may be applied. In this variant, the differences between successive attribute values are calculated. If all available spectral bands are extended with AP or DAP, long feature vectors will be generated. This may lead to computational difficulties when using small computers.

1.2   Topographic objects

Topographical objects to be recognized and mapped are numerous. Only a certain number of objects are displayed or printed in raster or vector format at a selected scale. This book focuses on two-dimensional (2D) mapping at map scales larger than 1:25000. The topographic maps will be used for planning, navigation, and as base for supplementing objects of other themes. Typical topographic objects are buildings, roads, trees, bushes, hedges, grass, lamp posts, bicycle paths, rivers, and coast lines. Non-topographic objects are cars and shadows, which must be removed from a preliminary processing result. The characteristics of objects are height, area, shape, and texture. The representation of objects using lines, polygons and symbols must be of high cartographic quality. The permanent objects, e.g., buildings and roads, demand high geometric accuracy. For these objects, a submeter accuracy is required. There is a significant need to increase the map production and the updating of the maps in the world. According to Konecny et. al. (2016) only 30 % of the world land area is mapped in scales 1: 25000 or greater and in some countries the time interval for updating is between 10 to 30 years.

1.3   Data sources

The source data may be aerial or satellite imagery, ortho-images, and digital elevation models. Their characteristics are presented in the following.

1)   Aerial imagery

Primary data source are aerial images taken by photogrammetric cameras. Advanced models have a large format, high geometric and radiometric resolution, a panchromatic channel (PAN), and four spectral channels (Red, Green, Blue, Near Infrared). Cameras used for topographic mapping also have devices for recording the position and attitude of each exposure and for enabling large and constant overlap of successive images. Metric cameras are calibrated. When applying the derived camera parameters, spatial objects can be reconstructed from two or more images. Figure 3 depicts some aerial cameras and some of their characteristics. Imagery taken from airplanes may have a ground sampling distance (GSD) of a few centimetres, longitudinal overlaps of 60-80%, and lateral overlaps of 20-60%. Images of the large-frame DMC III, e.g., have 375 million pixels (MP). At GSD=0.1m one image will cover an area of 2573m x 1459m or 3.75 km^2 on the ground. The colour bands are generated by additional lenses. High-resolution colour images are then generated by combining a high-resolution panchromatic image with images of the spectral bands of lower resolution. This technique is called pan sharpening. Different techniques exist. At IHS pan sharpening, e.g., the low resolution RGB bands are first up sampled to the same resolution as the panchromatic band and then transformed into the I(ntensity), H(ue), S(aturation) colour space. The I-channel is then substituted by the high-resolution panchromatic channel and a back-transformation produces a high-resolution colour image with RGB bands (Albertz, 1991). The ratio between the pan- and the colour- resolution is, e.g., at DMC III 1:3.1. The dynamic range of the DCM III camera is 75 decibels (dB). High image quality is achieved by compensating for the forward motion of the airplane during the exposure time.

UltraCam-Xp   UltraCam-Condor

High-resolution and multi-spectral   DMC III   RCD30

Figure 3: Aerial photogrammetric cameras.

Overlapping imagery can also be taken from medium-format cameras. Due to their lower size and weight, they may be installed in unmanned airborne vehicles (UAV). Such cameras have sensors of about 60 MP as for example at the RCD30 camera. At GSD=0.1m one RCD30 image covers an area of 896m x 671m or 0.60 km^2 on the ground. The RGB colours are produced by a Bayer array and the dynamic range is 73 dB. More details on medium-format cameras and their use in topographic mapping are published in (Höhle, 2012).

An important property of a digital camera is the dynamic range (DR) of its sensor. It is the ratio between the largest and smallest intensity value also named digital number (DN). DR is derived by equation (3).

$$DR = 20 \cdot log_{10}(2^{Q}/1) = 6.02 \cdot Q \ [dB] \tag{3}$$

where:

Q = number of bits for DN_{max} , dB = decibel.

In case of the DMC III camera, the maximum digital number (DN_{max}) is $2^{12.96} = 7968$. A high DR enables the detection of details in areas with shadows. For the coding of the numbers to binary format, this camera uses 14 bits or 16384 DN. The processing of the images may be carried out with a lower range, e.g., 8 bits or 256 DN to avoid big amounts of data. The transformation of the original range to a range of 256 DN is done by an algorithm which keeps the important information.

Another important property is the instant field of view ($IFOV\_sensor$) of the sensor. It is defined in (Sandau, 2005) by Equation (4).

$$IFOV_{sensor} = pel \ / \ c \ = 1000 \ / \ k \ [mrad] \tag{4}$$

where:

c = camera constant
pel = pixel size

For the DMC III: $1/k$ = 0.0039/92 = 0.0000423913 or 0.0423913 mrad.

The flying height (h) above ground for a required GSD is calculated by equation (5).
$$h = k \cdot GSD \tag{5}$$

2) Satellite imagery

The use of satellite imagery is also a possibility for source data. It has the advantage of covering large areas quickly and repeatedly. The images may have a higher spectral resolution, but their constant ground resolution is lower than that of aerial cameras. The WorldView-3 satellite, for example, captures overlapping panchromatic images with GSD = 0.31 m and 8 spectral channels. Pan-sharpened images are available with a radiometric resolution of 8 or 16 bits. The satellite images can be taken across national borders, for example the annual collection of land use data for all of Europe. The logistics involved in such a task are simpler than operating an aircraft.

3) Ortho-images

Ortho-images are produced from the original images either in natural color or infrared color (CIR) or black and white. They are corrected for camera tilts during exposure and changes in elevation of the terrain. They may be georeferenced, but the pixel size can be different from the GSD. There are two types of ortho-images. One is based on a digital surface model (DSM), the other on a digital terrain model (DTM). The properties of the DSM-based ortho-image are a uniform scale at all positions of the ortho-image. In areas with great changes in height, e.g., at the edge of buildings, the image quality is, depending on the density and quality of the DSM, deteriorated. The outlines of the buildings are not straight and continuous anymore. At the DTM-based ortho-images, only the objects on the terrain have a correct position.

4)   Elevation data

Elevations can be derived from overlapping images or by laser scanning. When using image matching, a point cloud of high density and accuracy can be derived. A digital surface model (DSM) is obtained by interpolating elevations for a grid or triangles. The DSM may be modified by identifying ground points and deriving grids or triangles from them. The result is a digital terrain model (DTM) which represents the ground (bare earth). Both models may be used for the generation of ortho-images or attributes to be used in classification. The difference between DSM and DTM is a normalized DSM (nDSM) representing heights above ground. The advantage of laser scanning lays in the fact that ground points are better detectable in areas of dense vegetation. In recent years, laser scanning has been used for the generation of nation-wide DTMs. In the following we focus on the DSM and its derivation from imagery.

The achievable accuracy depends on many factors which are related to terrain, camera, flight parameters, image characteristics, processing, and assessment methods. It seems to be impossible to have a simple answer to the question which is the best instrumentation and procedure for DSM/DTM generation. In Table 1 results of three investigations are presented.

Table 1: Results of DSM/DTM evaluation. (*RMSE*=root mean square error, *N/A*=not applicable/available, *IFOV_sensor*=instant field of view of sensor, *ALS*=airborne laser scanning, *s_fd*=image side in flight direction, *p*=overlap in *fd*, *q*=overlap across *fd*, *b*=building, *s*=impervious surface/sealed, *g*=grass, *t*=tree, *b*=building).

Source	Author(s)	Höhle	Haala et al.	Ressl et al.
	Year	**2009**	**2010**	**2016**
Terrain	landscape	open	open	open & built-up
	landforms	smooth	smooth	smooth
	Δh	<30 m	N/A	N/A
	average slope [%]	10	N/A	N/A
	land categories	g, b, t, s	s	g, s
Camera	model name	UCD	DMC	UCX
	focal length [mm]	101	120	100
Sensor	pixel size [µm]	9	12	7.2
	IFOV [mrad]	0.089	0.100	0.072
Image	s_fd [pel]	7500	7680	9420
	s_across_fd [pel]	11500	13824	14430
Flight	GSD [cm]	6	8	6
	p/q [%]	58	80/60	80/70
Processing	program	ISAE_v5.1	M-T-DSM	M-T-v5.5 (UAS)
	feature	single model	multi-model	multi-model
	grid size [m]	5	0.20	0.15
Reference	point types	mass	signalized	mass
	method	GPS/RTK	GPS/static	ALS
	# of check pts	366	60	1244494
	density [pts/m^2]	N/A	N/A	4

Assessment		3·RMSE	3·RMSE	-0.8/2.5·RMSE
	threshold	3·RMSE	3·RMSE	-0.8/2.5·RMSE
	RMSE [cm]	14	3.3	12.1
	RMSE [GSD]	2.33	0.41	2.02
	mean [cm]	1.0	0.9	10.8
	std. dev. [cm]	13	N/A	5.8
	std. dev. [GSD]	2.16	N/A	0.97

The results of Ressl et al. (2016) show that the derived mass points have an accuracy of $\sigma_h \approx 6$ cm or one GSD. The $RMSE_h$, however, is about doubled due to a large area with grass. The image matching determines the surface of such areas while ALS determines the terrain elevation. Signalized points on impervious ground are determined with higher accuracy, e.g., with $RMSE_h=3.3$cm in (Haala et al., 2010). The results show also the progress achieved over the years. Advancement in cameras and processing techniques are responsible for, that the obtainable results for automated generation of elevations from aerial imagery are of high-quality regarding density, completeness, and accuracy, provided that the images have contrast and structure. Advanced processing methods use semi-global matching (SGM) and multiple stereo-models of varying overlaps.

Elevation data may also be acquired by airborne laser scanning (ALS). Its advantage is that the terrain may be determined with higher accuracy, especially at vegetated areas. For example, in the investigations of Ressl et al. (2016) the DTM has been determined with $\sigma_h \approx 4$ cm. For the generation of DSM, however, the good 'penetration behaviour' of ALS may be a disadvantage. Both methods may produce gross errors which have to be detected and eliminated. The obtainable accuracy depends very much on the terrain if it is vegetated, sealed, built-up, textured, or well reflecting. It is a matter of fact, that not all gross errors can be detected. For the assessment of the accuracy robust statistical measures, normalized median absolute deviation (NMAD), and 68% Quantile (Q_{68}) of the absolute differences, must be used. More details on these accuracy measures may be found in (Höhle & Höhle, 2009).

DSM derived from satellite imagery is investigated in (Gong and Fritsch, 2019). The achieved accuracy with overlapping WorldView-3 satellite imagery is quoted with NMAD = 0.7 m or 2.4 GSD and Q_{68} = 1.0 m or 3.4 GSD (average of three test sites). ALS data were used as reference.

References

Albertz, J., 1991. Grundlagen der Interpretation von Luft- und Satellitenbildern. Wiss. Buchges. 1991, ISBN 3-534-07838-1.

Dalla Mura, M., Benediktsson, J. A., Waske, B., and Bruzzone, L., 2010. Morphological attribute profiles for the analysis of very high-resolution images. IEEE Transactions on Geoscience and Remote Sensing, 48(10), 3747-3761.

Damodaran, B., Höhle, J., and Lefévre, S., 2017. Attribute profiles on derived features for urban land cover classification, Photogrammetric Engineering & Remote Sensing, vol. 83, no. 3, 183-193.

Gong, K. and Fritsch, D., 2019. DSM Generation from high resolution multi-view stereo satellite imagery. Photogrammetric Engineering, vol. 85, no. 5, 379-387.

Haala, N., Hastedt, H., Wolff, K., Ressl, C., Baltrusch, S., 2010. Digital photogrammetric camera evaluation – generation of digital elevation models, PFG 2010 (2): 99-115.

Höhle, J., 2012. Medium-format cameras and their use in topographic mapping, ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. I-4, 77-82.

Höhle, J., 2009. DEM generation using a digital large format frame camera. Photogrammetric Engineering & Remote Sensing, vol. 75, no. 1, 87-93.

Höhle, J. and Höhle, M., 2009. Accuracy assessment of digital elevation models by means of robust statistical methods. ISPRS Journal of Photogrammetry and Remote Sensing 64:398-406.

Konecny, G., Breitkopf, U., Radke, A., 2016. The status of topographic mapping in the world. A UNGGIM - ISPRS project 2012–2015. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLI-B4, 737–741.

Pham, M.-T., S., Lefèvre, S., Aptoula, E., 2018. Local feature-based attribute profiles for optical remote sensing image classification. IEEE Transactions on Geoscience and Remote Sensing, 56(2), 1199.

Ressl, C., Brockmann, H., Mandlburger, G., Pfeifer, N., 2016. Dense image matching vs. airborne laser scanning – comparison of two methods for deriving terrain models. PFG 2016 (2): 57-73.

Sandau, R., 2005. Digitale Luftbildkamera, ISBN 3-87907-391-0, Herbert Wichmann Verlag.

CHAPTER 2. Classification

2.1 Objective

The objective of remote sensing image classification is to convert the aerial/satellite image into a raster map automatically with labels for the selected topographic objects. For each unit of the map, a class label must be assigned, which is accomplished by an algorithm, also named as classifier. The classes are the topographic objects, e.g., buildings, roads, trees, bushes, etc. In this chapter, we will discuss in detail about how such a classifier can be designed.

2.2 Machine Learning

Machine Learning can be considered as the black box, which takes some inputs and outputs the meaningful information. For example, in terms of classification the black box takes inputs as the pixel intensities of the image, and outputs whether it belongs to some specific land cover classes. Here the black box is called, as the classifier. The objective of the machine learning is to learn such a classifier from the underlying data (image) to perform classification. Depending on how the classifier is learnt, it can be categorized into Supervised Classification, and Unsupervised Classification. The description of the unsupervised classification is out of the scope of this chapter, so here we focus on the supervised classification.

2.2.1 Supervised Classification

In supervised learning (classification), we have access to the inputs and its corresponding outputs. Let $\{(x_1, y_1), (x_2, y_2), \ldots, (x\_n, y\_n)\}$ be the pairs of inputs and outputs, and our goal is to find/learn a classifier f which maps the inputs to the outputs with minimal error. In the case of remote sensing, let us assume that we have the satellite image and its corresponding topographic map indicating the object labels, which is also called as ground truth map. Figure 4 shows the remote sensing image and its corresponding ground truth map which consists of six different land cover classes/objects. In certain cases, the ground truth map may be partially or sparsely labeled through the field work. In this case, our objective is to learn the classifier using the ground truth map/samples collected from the field survey.
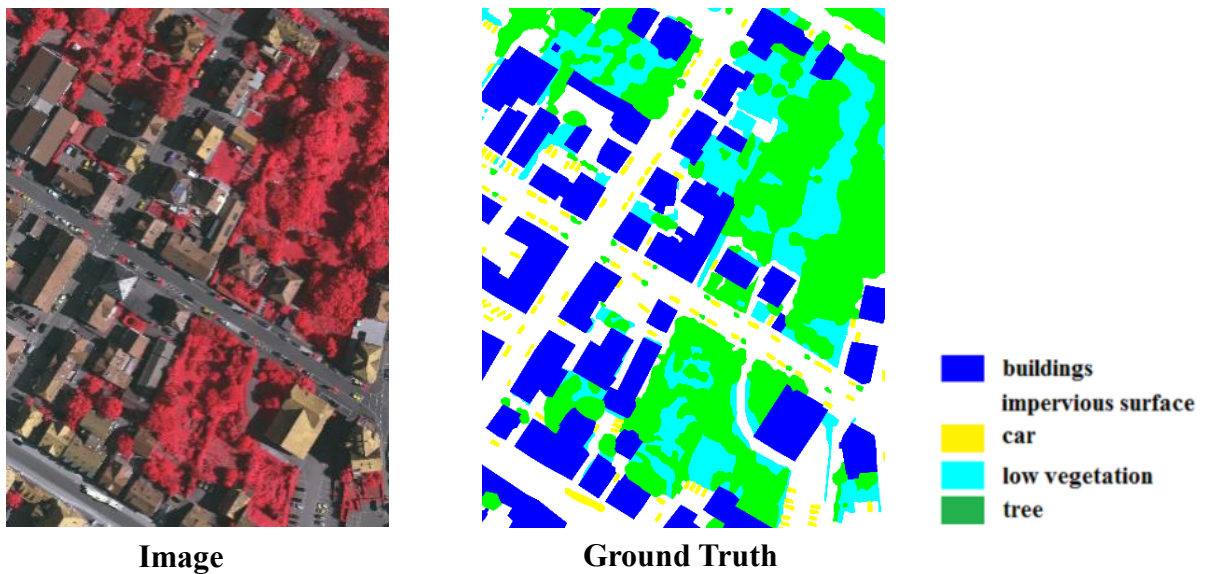
Image **Ground Truth**

Figure 4: Aerial image from the ISPRS dataset displayed in the False color composite. On the right shows its ground truth map which indicates the labels of each pixel in the aerial image.

The training samples to train the classifier is constructed from the image and ground truth map as shown in Figure 4. The columns indicate the pixel values in the image at a particular location, and the last column indicates class labels of the pixel. In machine learning literature, the pixel values along the channels, that is columns are called as *"features"*, and the rows are called as "*samples*" (each pixel at a particular location). The number of rows indicates the total number of training samples, and total number of columns indicates the number of features (without considering the class label column).

Band 1	Band 2	Band 3	Band 4	labels
0.252	0.464	0.312	0.785	1
0.425	0.464	0.535	0.215	2
0.312	0.656	0.231	0.697	3
0.101	0.322	0.462	0.323	4
0.162	0.402	0.621	0.105	5
0.272	0.434	0.326	0.775	1
0.115	0.312	0.442	0.353	4
0.405	0.484	0.525	0.205	2
...	...	...	...	...

Ground Truth

buildings
impervious surface
car
low vegetation
tree
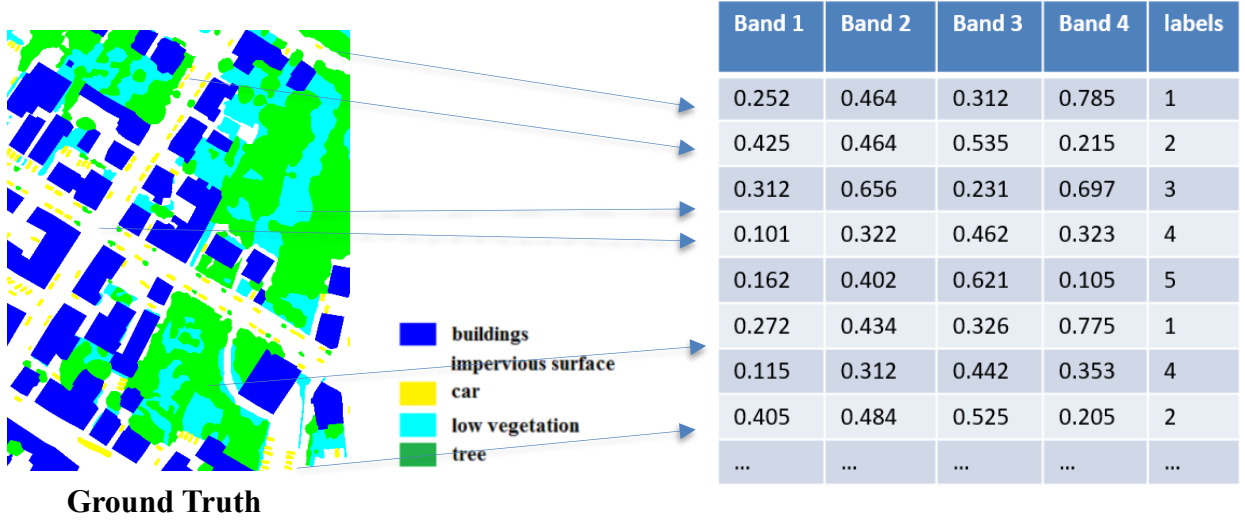
Figure 5: Ground Truth map and training samples to learn the classifier. The table on the right shows the pixels values of the 4 channels (blue, green, red, NIR), and its corresponding class labels, where *"1"* indicates the class *"trees"*, *"2"* for *"car"*, etc.

Once we have generated the training samples, the question remains that how do we learn the classifier such that it takes the pixels values as the input and outputs the class label index?

2.2.2 Empirical Risk Minimization

The objective of supervised learning algorithm is learn a classifier (or a function):

$$f : X \rightarrow y \Rightarrow f(X) = y, \tag{6}$$

here we call *f(X)* as the predicted value of classifier *f* and *y* is the actual value to be predicted. The value of *y* is known to us, because *(X,y)* is the training pairs as shown in the Table of the Figure 5. Now the idea is to find *f* or learn *f* such that it's predicted value is as close to *y*. In other words, the *error = f(x)-y* should be minimum.

In order to measure, how well a function or classifier fits a training data (X_i, y_i), we define a loss function $L(y_i, f(X_i))$. The loss function is high when the predicted value of the classifier deviates from the ground truth value, and it's low when it predicts closer to the ground truth value. The loss of the classifier over the entire training data can be defined as:

$$E = \frac{1}{N} \sum_{i=1}^{N} L(y_i, f(X_i)), \tag{7}$$

where N is the total number of training samples, and E is the average loss over the training dataset. One can guess that we need to learn a classifier which minimizes the average loss E, this is known as the empirical risk minimization. Depending on the design or choice of the loss function, the classifier changes. One simple loss function is the mean square error. In order to minimize the loss function E, the classifier rely on the optimization method, and depending on how they minimize the loss function or how they model the classifier we have different classification methods. In the rest of the chapter, we discuss about the well-known classifiers in the remote sensing community.

2.2.3 Classifiers

In the remote sensing literature, there exists several classifiers depending on the formulation. It is noted that there is no single best classifier which performs well across different images or applications. The choice of the classifier depends on the nature of the problem and available number of training samples. In this book, we focus on the four machine learning methods:

1. Decision Tree (DT)
2. Random Forest (RF)
3. Support Vector Machine (SVM)
4. PerTurbo classifier.

These classifiers are selected because of its potential to tackle the problems in the remote sensing image classification. Typically, in remote sensing we end up with two scenarios:
   (a) large scale dataset, where the number of the training samples are very large
   (b) small number of the training samples, where it is difficult or expensive to collect the ground truth map through field study.

The first two classifiers (DT and RF) more suited to tackle the former problem, while the remaining two classifier are suited to the later small training set problem. The later small number of training samples problem are also referred as Hughes Phenomena, where the number of training samples are small compared to the dimension of the dataset.

2.2.3.1 Decision Tree

The decision tree classifier is very straight forward. It asks a series of questions and conditions in tree structure, as shown in the Figure 6, for which the answer is yes or no. In other words, it is a binary.
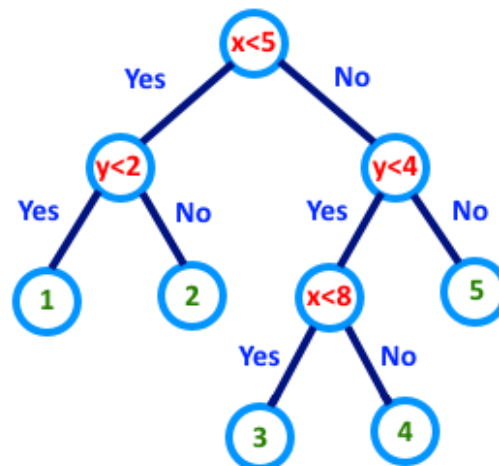

Figure 6: Decision Tree (DT) classifier construction.

Let us consider the training samples table in Figure 5. Here we have four features: band 1, band 2, band3, band 4, these are also called as attributes. The decision tree is constructed as follows:

1) Pick an attribute and formulate the question. For example, let us pick band 4, and check the condition whether the band $4 < 0.22$

2) We split the table into two, the samples which agrees the condition band $4 < 0.22$ are on the right, and otherwise it is on the left

3) Now we choose another attribute or feature, band 2, and check the condition for whether band $3 > 0.5$, and split the tables into two accordingly to the answer

4) Repeat this process, until the partition belongs to the same class label. In other words when the homogeneity is achieved.

Band 1	Band 2	Band 3	Band 4	labels
0.252	0.464	0.312	0.785	1
0.425	0.464	0.535	0.215	2
0.312	0.656	0.231	0.697	3
0.101	0.322	0.462	0.323	4
0.162	0.402	0.621	0.105	5
0.272	0.434	0.326	0.775	1
0.115	0.312	0.442	0.353	4
0.405	0.484	0.525	0.205	2
...	...	...	...	...

Figure 7: Training samples table (from Figure 5)

In general, it is difficult to formulate the question by hand. So the selection of the attribute and splitting is done automatically. The selection of the attributes are based on the information gain, *'Gini index'*, which measures how pure is the separation. If the separation is homogeneous, then the *'Gini index'* is small, thus it can be used to selection of the variables and also to determine the threshold to split the attribute into two. *Gini index* is calculated for the range of thresholds, and the threshold which has lowest *Gini index*, is selected for splitting the attributes. This process is called as the decision trees. The tree is then pruned, based on some criteria that each partition must contain certain number of samples or else it is merged with its former partition. The theoretical background of the DT method is given by Breiman et al. (1984).

The advantage of the decision tree classifier is that it scales well for the large-scale dataset or big data. The process can be computed in the parallel. However, the main disadvantage is that it is more sensitive to the change in the feature values. For example, if there is a small change in the feature values, the output can be changed drastically, thus it has high variance in its prediction. The other issue is that DT overfits the training data, and has less generalization performance. That is the training accuracy is very high, whereas the testing accuracy might not be high as the training accuracy.

2.2.3.2 Random Forest

Random Forest is an ensemble classifier, which uses decision trees to construct the base classifiers in the ensemble learning framework (Breiman, 2001). The combination of the several decision trees makes the RF as a powerful classifier which can reduce the variance in the predictions and scale well to the large-scale dataset. For this reason, this classifier has been widely used in the remote sensing. Let us consider we have N number of samples and M number of features or attributes. The construction of the RF is as follows:

1) Randomly choose K number of samples from N number of samples with replacement

2) Randomly choose P number of features/attributes from M number of features

3) Build the decision tree classifier

4) Repeat the step from 1 to 3, until the desired number of classifiers L is achieved

5) Perform majority voting on the output of the L decision tree classifier.

The schematic scheme of the random forest is shown in Figure 8. Here X is the dataset, and only subset of features is randomly selected for each decision tree classifier. Thus in certain cases, all the training samples might be used and only the subset of features in each decision tree. This choice depends on availability of training samples, when the number of training samples are very large then the random selection of the training samples mentioned in step 1 is necessary.
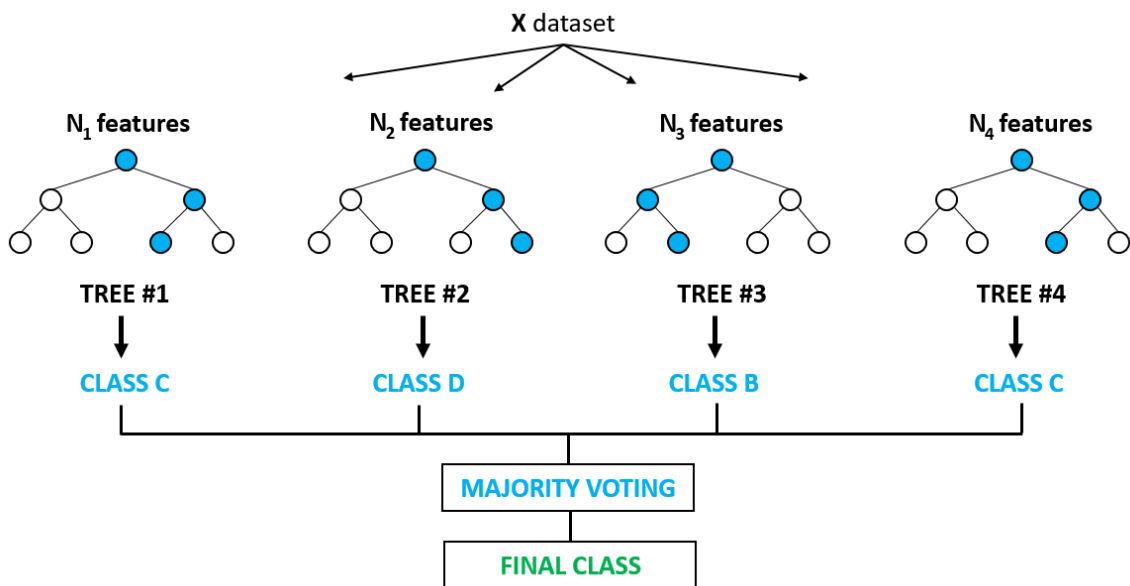
Figure 8: Random Forest classifier. Each decision tree classifier is constructed using the randomly selected attributes. The final decision is performed by majority voting.

The final decision of the RF is performed by majority voting. For example, in the Figure 8, the pixel is classified into class 'C' by two decision tree, 'B' and 'D' by one each decision tree. Thus, the pixel is assigned into class 'C'. When there is a tie, then the pixel is assigned into the class label randomly. Apart from the classification, RF also ranks the importance of the features/attributes which can be used to feature selection problem in the high dimensional dataset.

2.2.3.3 Support Vector Machines

Support vector machines (SVM) is a binary classifier, the goal of the SVM is to find the hyperplane such that it can separate into two classes. SVM identifies an optimal separating hyperplane which maximizes the margin between two classes of the training data. In two dimensions, the hyperplane is line, and high dimensional the separation is the hyperplane. Let us consider the following 2D data (Figure 9) which consists of red and blue points, the goal is to draw a line which separates the red and blue points. There are several possible lines which can separate the two regions (on the right side of Figure 9), now the goal is to find the best separating hyperplane.
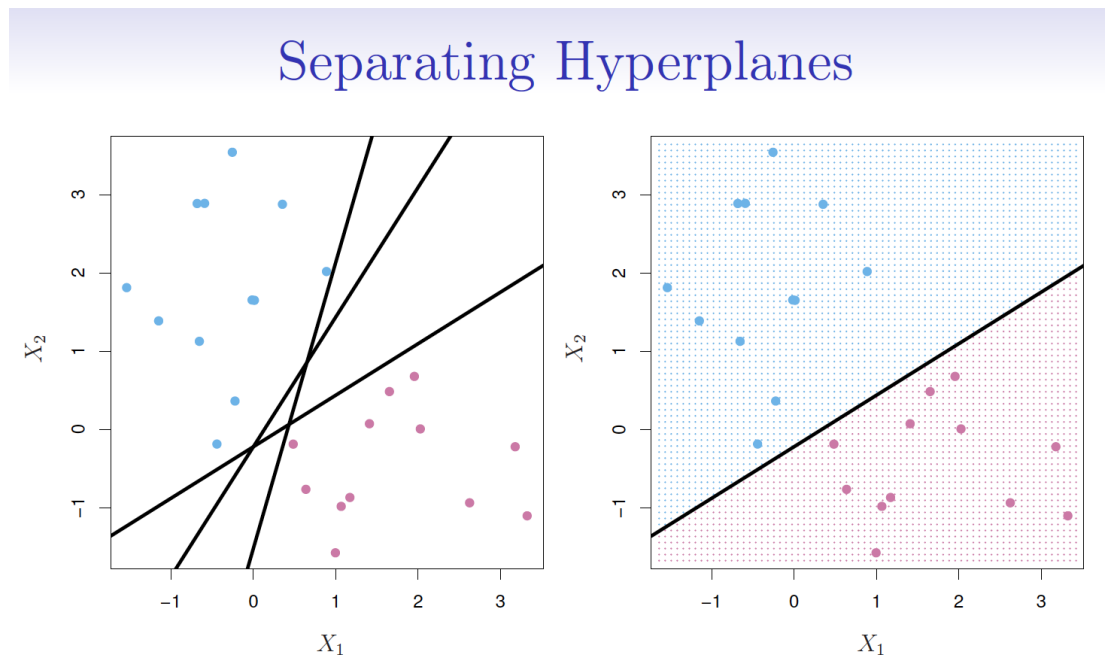
Figure 9: The classification hyperplane between the red and blue points. There are several hyperplanes possible to separate the two regions.

The best hyperplane is the one that lies far away from both blue and red points (extreme blue and red points). In other words, the best hyperplane is the one that maximizes the margin.

- The margin is the distance between the hyperplane and a few close points.
- These close points are called **support vectors** because they control the margin.

The solid line in Figure 10 is the one that maximizes the margin, and this is known as *support vector machine classifier*.
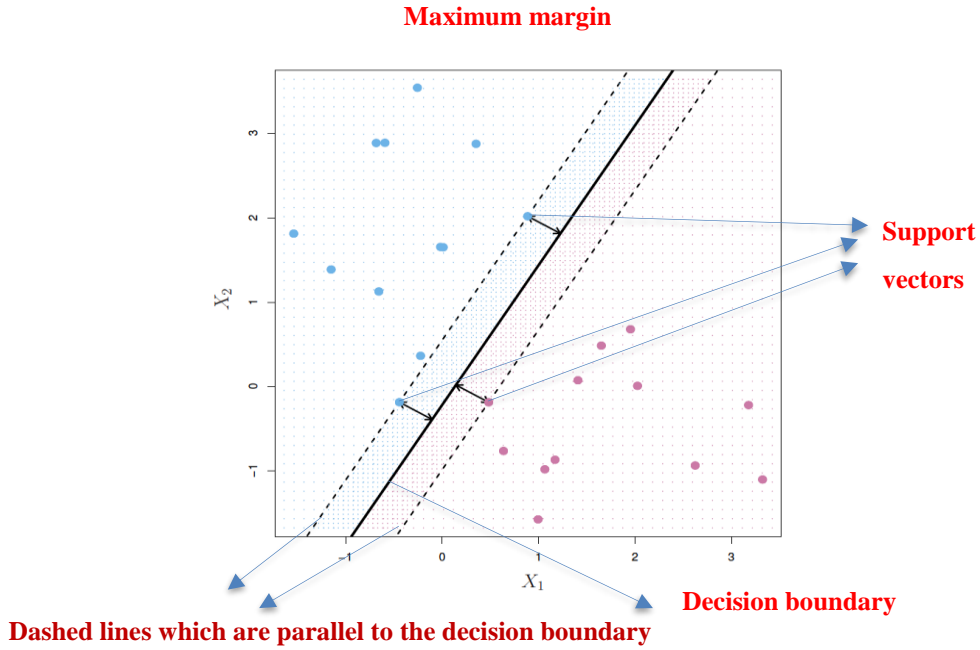
Figure 10: Decision boundary of the support vector machines classifier.

Now scenario arises, what happens, if the blue and red points are not perfectly separable, that is there are few samples/points which is noisy. This is the case in most real-world cases (as shown in Figure 11). In this case, the classifier defined in Figure 10 might not be the optimal one. In order to include the noisy data points, the SVM is reformulated to allow some misclassifications through the soft margins. The constraint of maximizing the margin of the line that separates the classes must be relaxed. This is called the soft margin classifier. This allows some points in the training data to violate the separating line. A tuning parameter is introduced called C which defines the amount of violation of the margin allowed. The value of C is specific to the underlying dataset. This tuning parameter C is important when looking to minimize error. When the C is small, the classifier allows only a small bit of misclassification. We may overfit the training data if our tuning parameter is too small. When C is large, the number of misclassifications allowed has been increased. This classifier would generalize better but may have a high amount of bias. When C is zero, there can be no misclassification. This is depicted in Figure 12.
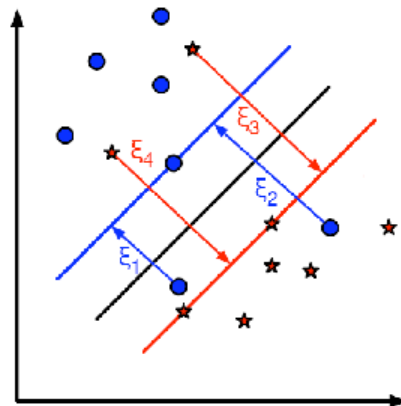
Figure 11: SVM separating hyperplane when the data points are not exactly linearly separable. The two blue points lies in the region of the red region and vice versa.
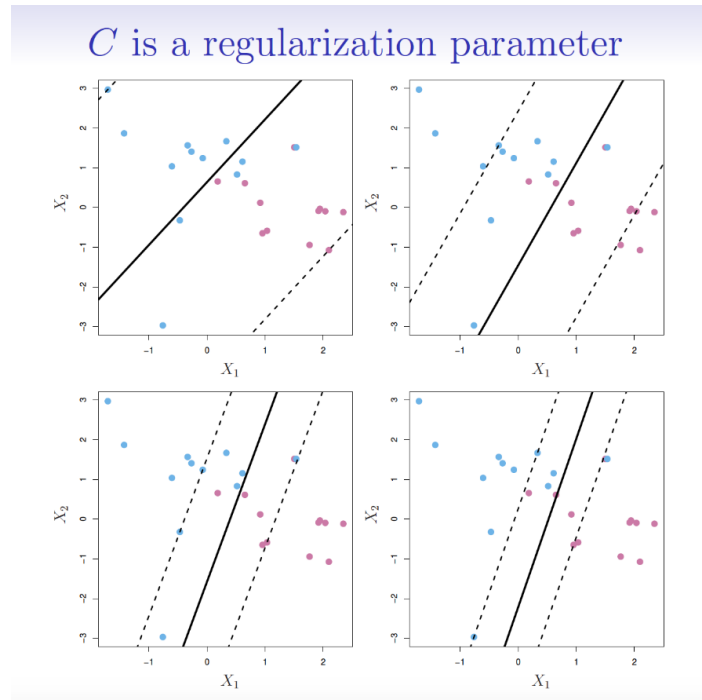
Figure 12: Classification hyperplane with different values of C

Figure 12 shows classification boundary with different values of for the different configurations/values of C. The best C is determined by the cross-validation.

In the most of the cases, the data we are dealing might not be linearly separable, that is a line in 2D cannot separate two regions. In this case, we might need a curve to separate the two regions. Thus in order to take advantage of the linear separation formulation of the SVM, the data is transformed into higher dimensional space, where the data is linearly separable. Should we need to add this feature manually to have a hyper-plane? No, SVM has a technique called the kernel trick. These are functions which takes low dimensional input space and transform it to a higher dimensional space i.e., it converts not separable problem to separable problem, these functions are called kernels. Figure 13 shows an example which transforms non-separable 2D points in the original space into the linearly separable in the 3D space using polynomial transformation.
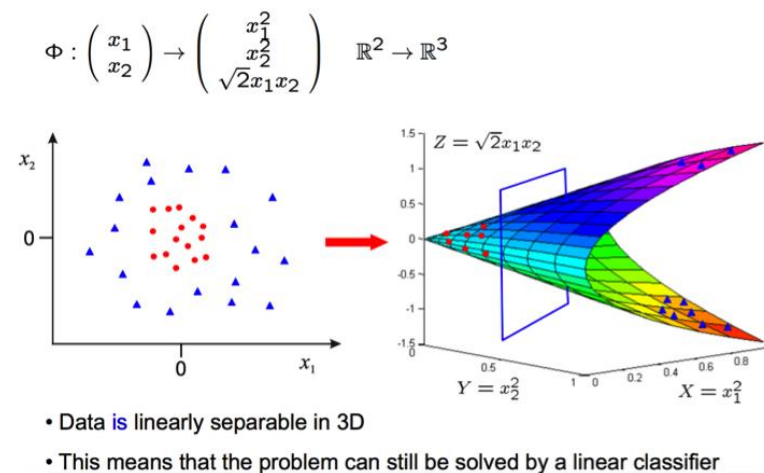
Figure 13: Kernel trick, the polynomial mapping of data into high dimensional space to make the data as linearly separable.

The kernel matrix is computed based on the similarity between the samples in the dataset. There is several types of kernel, linear, polynomial, radial basis function, and sigmoid. However, the widely used kernel is radial basis function (RBF) or Gaussian kernel due to its superior performance. The RBF kernel is defined as:

$$K(x_i, x_j) = exp(-\gamma \| x_i - x_j \|^2), \gamma > 0, \tag{8}$$

γ is called as band-width parameter, which must be tuned to underlying dataset.

2.2.3.4 PerTurbo classifier

PerTurbo is a discriminative classifier in which each class is characterized in the manifold by the Laplace-Beltrami operator (Chapel et al., 2014). The Laplace-Beltrami operator is approximated using the Gaussian RBF kernel, which we have seen in the SVM classifier section. Here for each class, we compute the kernel matrix, instead of computing for the entire dataset. Thus, if there are M classes, we compute M kernel matrix for each class. Once class specific kernel matrix are computed, then when the new sample comes it is projected on to each classes kernel matrix, and the sample is different from the kernel matrix, it outputs large value. We call it as a perturbed value. Thus, if the perturbation is higher, then sample does not belong to that particular class. The new sample is classified into the class, which has smallest perturbation among all the classes. The PerTurbo classifier contains two parameters to be optimized, the bandwidth parameter of the Gaussian RBF kernel and a regularization parameter for determining the inverse of Gaussian RBF kernel. These hyper-parameters are automatically tuned using a cross-validation approach. The PerTurbo is classifier is very simple to train, as it only requires the computation of the kernel matrix in the training time.

2.2.4 Conclusion

Now we have seen four types of classifiers. As mentioned earlier, the suitability of the classifier depends on the underlying problem. The decision tree, and random forest classifier are mostly well-suited to handle large scale datasets, whereas the SVM and PerTurbo are well-suited to small sample size problem, especially in the case of "curse of dimensionality", where the number of samples are small compared to its dimension. It arises in the case of hyperspectral images, are in the cases of the large number of features (for example, attribute profiles).

References

Boehmke, B. and B.M. Greenwell, 2019. Hands-On Machine Learning with R. CRC Press, ISBN 9781000730197.

Breiman, L., J. Friedman, C.J. Stone, and R.A. Olshen, 1984. Classification and Regression Trees, CRC Press.

Breiman, L., 2001. Random Forests. Machine Learning, 45(1):5–32.

Chapel, L., T. Burger, N. Courty, and S. Lefèvre, 2014. PerTurbo manifold learning algorithm for weakly labeled hyperspectral image classification, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 7(4):1070–1078.

Melgani, F., and L. Bruzzone, 2004. Classification of hyperspectral remote sensing images with support vector machines, IEEE Transactions on Geosciences and Remote Sensing, 42(8):1778- 1790.

CHAPTER 3. Assessment and enhancement of classification maps

3.1 Introduction

This chapter covers the techniques to assess the quality of the extracted topographic data. As the urban landscapes consist of objects with different shapes, we cover methods not only related to thematic accuracy but also to the geometric accuracy of the extracted and enhanced topographic map data. Depending on the results and the demands, it might require some additional processing techniques to enhance the quality of the classification maps. The techniques comprise algorithms like 'Hough transform' and "Least-squares adjustment" for lines, rectangles, and polygons. The processing is carried out semi-automatically. Open-source software and documentation can be accessed at *https://github.com/JoaHoe/buildenh_jh*.

3.2 Assessment of thematic accuracy of land cover maps

The final maps must meet certain quality standards. This is true also for the intermediate steps. Therefore, the generated land cover maps must be tested for their quality. The checking of the thematic accuracy requires reference values which are compared with the generated values at the same position. This can occur pixel-wise or object-wise. It is not necessary to test each pixel or object. In most cases, the reference values are collected for some samples only. Their size must be derived by statistical methods.

Size of the sample

A simple formula for sample size (*n*) is given in (Congalton and Green, 2009).

$$n = B / 4 \cdot b^2 \qquad (9)$$

where:

n = sample size
B = χ2 (m=1, 1-α/k)
m = degree of freedom
1-α = desired confidence level
k = number of classes
b = desired precision of the entire sample.

The value for *χ2* may be taken from statistical tables or from the *R* function (*'qchisq'*). The meaning of the parameters is explained in Figure 14.
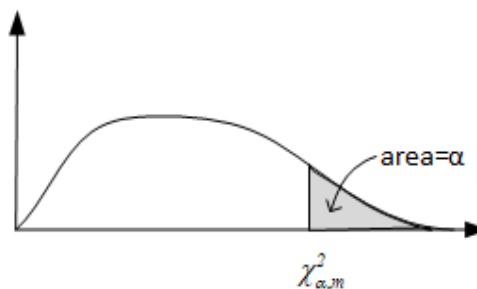
Figure 14: Chi-square density function (source: Mikhail, 1976)

At the formula it is assumed that the class area is 50% of the entire population (worst-case scenario). For the computation, a R-program is provided.

A more general solution for the sample size is published in (Höhle & Höhle, 2013). The corresponding programs are described in the Appendix at Module 4 / software tools. The scripts can be downloaded from Moodle.

The thematic accuracy of the generated classes is checked by comparing the reference value with the generated value. The results of the comparison are presented by an error matrix from which the accuracy measures are calculated (cf. Figure 15). The rows represent the classified data and the columns the reference data. The counts in the diagonal of the matrix are agreements and the counts off the diagonal are errors.

		j=columns (reference)			row total n_{i+}
		1	2	k	
	1	n_{11}	n_{12}	n_{1k}	n_{1+}
i=rows (classification)	2	n_{21}	n_{22}	n_{2k}	n_{2+}
	k	n_{k1}	n_{k2}	n_{kk}	n_{k+}
column total n_{+j}		n_{+1}	n_{+2}	n_{+k}	n

Figure 15: Error matrix

The accuracy measures are overall accuracy (OA), producer accuracy (PA), and user accuracy (UA). PA is related to the omission errors and UA to commission errors. Formulas for the accuracy measures are published in (Congalton and Green, 2009).

$$overall\ accuracy = \frac{\sum_{i=1}^{k} n_{ii}}{n} \tag{10}$$

$$user's\ accuracy_i = \frac{n_{ii}}{n_{i+}} \tag{11}$$

$$producer's\ accuracy_j = \frac{n_{jj}}{n_{+j}} \tag{12}$$

where:

i = row number (classification),
j = column number (reference),
k = number of classes (categories),
n_{i+} = row total,
n_{+j} = column total.

Use of weights

The areas of the used classes are often quite different in size, so the use of weights gives a better answer for the thematic accuracy of the individual classes.

$$\overline{n}_{ij} = w_i \cdot n_{ij} \tag{13}$$

$$w_i = \frac{N_i}{N} \cdot \frac{n}{n_{i+}} \tag{14}$$

where:

$\overline{n}_{ij}$ = weighted count
N = total number units in the map
N_i = total number of units in the population of class i.

3.3 Enhancement of classification map

The generated land cover map is in raster format. A lot of deficiencies may be visible and require treatment. For example, areas are not homogeneous, lines are broken and wiggly, and objects are smaller than the specified minimum size. If the goal is to produce a final raster map, then the produced classification map still must be enhanced. Various image processing methods are applied. These procedures are described in (Höhle, 2017). In this book the focus is on the generation of vector data. The steps are described in the following sections. They correspond to the name of program package *"buildenh"* (cf. Section 'Software tools' in Appendix / Module 4).

3.3.1 Extraction of all classes

In a first step, the topographic objects (classes) are extracted one by one from the produced classification map. The enhancement to vector data is done for each class separately. The class "building" needs special treatment and will be dealt with in the following. The other objects (roads, hedges, vegetation) are treated by simpler methods and programs (cf. Section 'Mapping and assessment of other topographic objects' of this chapter).

3.3.2 Extraction of single buildings

The class "building" is extracted, enhanced, and the connected components (CC) are labelled using standard image processing methods (dilation, erosion, edge detection, segmentation, hull filling, feature extraction). Features of the CC (area, maximum radius) and of a fitted ellipse (centre, angle of major axis) are computed. These derived features support the following processing of each labelled building.

3.3.3 Detection of line segments

The detection of line segments in the CC representing one building uses the Hough transform where a line is expressed by Equation 15.

$$\rho = x \cdot \cos\theta + y \cdot \sin\theta \tag{15}$$

where

ρ = orthogonal distance of the line from the origin of the *xy*-system,
θ = azimuth of the normal vector to the line, and
x, y = pixel coordinates of the CC.

All pixels of the CC are transformed into the parameter space H (θ, ρ) using θ-values in the range between 0° and 175°. A proper resolution of H ($\Delta\theta$, $\Delta\rho$) must be selected. The ρ-range must be calculated so that all line segments may be detected. All pixels belonging to a line segment are accumulated in cells of the parameter space. The cell with the maximum number of counts is considered as the reference line (θ_{ref}, ρ_{ref}). The other lines are found by analysing the result of the Hough transform. The applied detection criteria are ρ_{min}, ρ_{max}, and n (length of line) for θ_{ref} and θ_{ref} +90°. Small lines may be found by manual digitizing a single pixel. A plot of all detected lines may discover whether the detected line segments describe the building correctly.

3.3.4 Determination of line sequence

The sequence of lines is determined by means of the angle between the centre of the CC and the centre of the cluster representing a line segment (cf. Figure 16). Ordering the angles according to their size will result in the correct sequence. The calculation of the minimum distance between the vertices will enable a solution at complex shapes. An alternative approach traces the outline of the CC and searches for the centres of the line segments. A graphic display of all lines may be applied for checking.
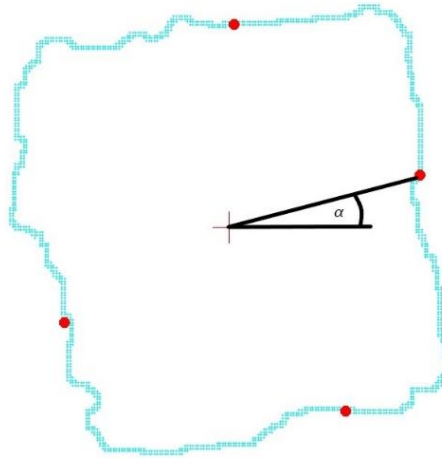
Figure 16: CC with centres of line segments (red dots) and azimuth (α).

3.3.5 Adjustment of lines

The parameters of the line are calculated by least-squares adjustment. The corresponding formulas are derived in (Borre, 1992). The orthogonal distance (r_i) from a point P(x_i,y_i) to a line (g) is given by Equation 16:

$$r_i = x_i \cos\theta + y_i \sin\theta - \rho \qquad (16)$$

The meaning of the parameters is depicted in Figure 17.
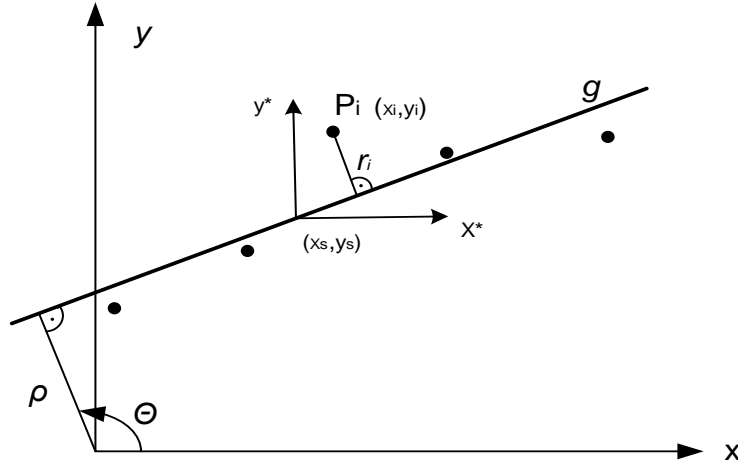
Figure 17: Orthogonal distance (r_i) from a point P(x_i, y_i) to a line (g).

Least-squares adjustment is achieved by minimizing the sum of r_i^2:

$$\sum r_i^2 = \min \tag{17}$$

The equations for the calculation of the two unknowns (θ, ρ) are:

$$x_s = \sum_{i=1}^{n} \frac{x_i}{n} \text{ and } y_s = \sum_{i=1}^{n} \frac{y_i}{n} \tag{18}$$

$$x_i^* = x_i - x_s \tag{19}$$

$$y_i^* = y_i - y_s \tag{20}$$

$$\tan 2\theta = \frac{-2\sum_{i=1}^{n} x_i^* y_i^*}{\sum_{i=1}^{n} (y_i^{*2} - x_i^{*2})} \tag{21}$$

$$\rho = x_s \cos \theta + y_s \sin \theta \tag{22}$$

There are two solutions for θ which differ by 90°. The correct solution is found by comparison with the approximate value gained by the Hough transform.

3.3.6 Intersection of corner points

Successive lines are intersected, and approximate corner positions are obtained by equations (23) and (24 - 27).

33

$$\begin{pmatrix} x_{P_k} \\ y_{P_k} \end{pmatrix} = \begin{pmatrix} q_1 & q_2 \\ q_3 & q_4 \end{pmatrix} \begin{pmatrix} \rho_i \\ \rho_{i+1} \end{pmatrix} \tag{23}$$

where:

x_{Pk}, y_{Pk} = coordinates of an intersected corner point,

q_i = coefficients containing trigonometric functions of successive angles (θ_i, θ_{i+1}) and

ρ_i = orthogonal distance of the line from the origin.

The coefficients (q_i) are determined by Equations (24 - 27).

$$q_1 = \frac{1}{\cos \theta_i} - \frac{\tan \theta_i}{(\sin \theta_i - \cos \theta_i \tan \theta_{i+1})} \tag{24}$$

$$q_2 = \frac{\tan \theta_i}{(\cos \theta_{i+1} \tan \theta_i - \sin \theta_{i+1})} \tag{25}$$

$$q_3 = \frac{1}{(\sin \theta_i - \cos \theta_i \tan \theta_{i+1})} \tag{26}$$

$$q_4 = \frac{-1}{(\cos \theta_{i+1} \tan \theta_i - \sin \theta_{i+1})} \tag{27}$$

3.3.7 Calculation of accurate corner coordinates

Accurate coordinates of the corner points are calculated in two steps. First, the weighted average of the angle (θ_{av}) is determined whereby the weights (w_i) are derived from the lengths of the lines.

$$\theta_{av} = \frac{\sum_{i=1}^{n} w_i \theta_i}{\sum_{i=1}^{n} w_i} \tag{28}$$

The second step is a least-squares adjustment which derives new values for the orthogonal distances (ρ_i) of the lines from the origin. Using vector and matrix notation the least-squares adjustment is expressed as:

$$Ax = b + r \tag{29}$$

$$\hat{x} = (A^T A)^{-1} A^T b \tag{30}$$

where:

A = design matrix representing the equations (24-27)
$\hat{x}$ = vector of the unknowns (ρ_i)
b = vector of approximate coordinates of the corners
r = vector of residuals

The standard deviation of the residuals (σ_r) is calculated by Equation (31).

$$\sigma_r = \sqrt{\frac{\sum\limits_{i=1}^{2m} r_i^2}{m}} \tag{31}$$

where

m = number of corners in the polygon.

The new values for the corner coordinates (p_i) can then be derived by multiplying the updated design matrix (A) with the calculated unknowns.

$$\boldsymbol{p} = \boldsymbol{A}\,\hat{\boldsymbol{x}} \tag{32}$$

The separation of the calculation of the two variables (θ, ρ) has the advantage that the linearization of the complex formulas is avoided and the equations for the coefficients of the matrix (A) are the same.

3.4 Assessment of the geometric accuracy of enhanced maps in vector format

The assessment of the geometric accuracy may occur by different methods. A quick overview about the quality of the result may be obtained when the vectors are overlayed with the reference.
A numerical validation is done by means of the overlapping area or displacements of the corner points.

3.4.1 Visual inspection

The reference may be an DSM-based orthoimage or an existing topographic map. In both cases the errors are visible. Figure 18 depicts an example. This may start an editing especially when deficiencies are observed during the processing of single buildings.

Figure 18: Enhanced outlines of buildings plotted on top of an DSM-based orthoimage.
(Source: Höhle, 2021)

3.4.2 Numerical assessment

Specifications and standards require numerical values which must be derived using accurate reference values. To derive numerical values for the geometric accuracy we may compare single points, lines, or areas of the generated objects and of the reference. Statistical rules must be applied to ensure the reliability of the results.

3.4.3 Point- (pixel-)wise checking

To access the geometric accuracy of the derived buildings, accurate checkpoints must be available. Suitable checkpoints are well-defined in the reference and in the enhanced classification. Suitable points are, e.g., the corners of buildings. Reference coordinates may be determined by means of the reference map or the DSM-based ortho-image. The standard accuracy measures are the root mean square error (RMSE), the mean (μ), and the standard deviation (s). They are determined for both coordinates (x, y).

$$\text{RMSE} = \sqrt{\frac{\sum \Delta^2}{n}} \tag{33}$$

$$\mu = \frac{\sum \Delta}{n} \tag{34}$$

$$s = \sqrt{\frac{(\Delta - \mu)^2}{n-1}} \tag{35}$$

where:

Δ = coordinate difference between reference and enhanced building,
n = number of checkpoints.

The reliability of the accuracy measures may be estimated by means of a confidence interval (CI). For example, the 95 % CI of the standard deviation is calculated by:

$$P\left\{ \sqrt{\frac{ms^2}{\chi^2_{1-\alpha/2,m}}} < \sigma < \sqrt{\frac{ms^2}{\chi^2_{\alpha/2,m}}} \right\} = 1 - \alpha \tag{36}$$

where:

P = probability
$m = n$ -1 (degrees of freedom)
$1 - \alpha = 0.95$ (confidence level)
s = standard deviation of sample
χ^2 = chi-square-distribution.

More information on the equation (36) can be found in (Mikhail, 1976). An R-program is available (cf. Section 'Software tools' in the Appendix/Module 4).

The number of checkpoints should be $n > 50$ and the accuracy of the checkpoints should be 0.1σ (Li et al., 2005). Gross errors may be eliminated. They are defined as $\Delta > 3\sigma$, their number (N_B) should be quoted. The planimetric accuracy may also be quoted as a distance between the reference and the corner of the enhanced building.

$$RMSE_d = \sqrt{RMSE_x^2 + RMSE_y^2}$$

(37)

An example of the point wise checking is depicted in Figure 19.

Figure 19: Distribution of checkpoints along with the buildings extracted from a manually compiled land cover map (ISPRS data set, area #1). (Source: Höhle, 2021)

3.4.4 Area-wise checking

The area-wise assessment checks a single class after the enhancement, e.g., buildings. The thematic accuracy assessment determines the rate of detected buildings. The buildings of the reference map are compared with the buildings of the enhanced classification map. When comparing with the reference, three categories of agreement are determined: True positive (TP), false negative (FN), and false positive (FP). Table 2 explains the meaning of the categories.

Table 2: Categories at comparison between reference and enhanced classification.

Category	Reference	Enhanced classification
TP	present	present
FN	present	not present
FP	not present	present

The following accuracy measures are derived using the formulae given in (Potůcková and Hofman, 2016).

$$completeness = \frac{||TP||}{||TP||+||FN||} \tag{38}$$

$$correctness = \frac{||TP||}{||TP||+||FP||} \tag{39}$$

$$quality = \frac{||TP||}{||TP||+||FP||+||FN||} \tag{40}$$

where:

$||\cdot||$ means number of buildings in one of the three categories.

A detected and enhanced building can be categorized as "true positive", when the overlap between the reference building and the automatically detected and enhanced building is better than a threshold (T). The overlap is calculated by:

$$overlap = \frac{A_{ref} \cap A_{det/enh}}{A_{ref}} \cdot 100\% \tag{41}$$

where:

A_{ref} = area of the building of the reference
$A_{det/enh}$ = area of the detected and enhanced building.

Figure 20 depicts the two representations of a building. The overlap area in the example amounts to 80% of the reference area.
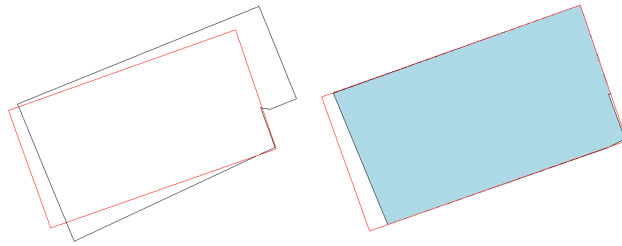
Figure 20: The detected/enhanced building (black line) and the reference (red line) at left and the overlap (blue area) on the right side.

At the numeric evaluation of the accuracy, the TP-, FN-, and FP-numbers are counted. For qualifying as TP, the overlap area between the enhanced building and the reference building should be larger than a threshold. In (Rutzinger et al., 2009) a threshold between T=50% and T=70% of the area of the reference object is suggested.

Other objects of the topographic database, e.g., impervious surfaces (roads, parking lots) and vegetated areas (grass), are composed of polylines. The segments may be straight or curved. The polylines can have an open end, or they can be closed. The formulas for the derivation of straight-line parameters are again based on the Hesse normal form, cf. Equation (15). The corner points (P_k) will be calculated by equations (23) and (24-27). The sequence of the lines is derived by ordering the calculated angles between the centre of the object and the corners.

In Figure 21, the closed polygon has five vertices (P_k). The segments of the polygon are straight lines (l_i) which are defined by the parameters (θ_i, ρ_i).
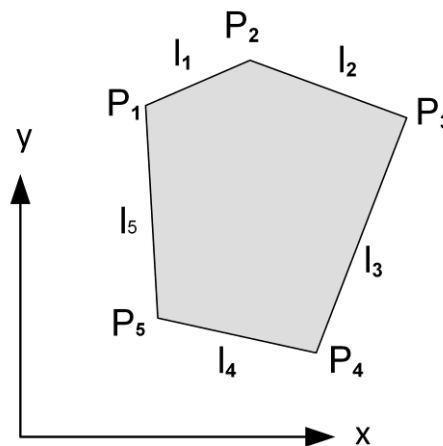
Figure 21: Closed polyline with five vertices (P_k) and straight-line segments (l_i).
(Source: Höhle, 2017)

A least-squares adjustment may be carried out using the constraint that the sum of the angles is a function of the number of corners.

References

Borre, K., 1992. Mindste Kvadraters Princip. ISBN 87-984210-1-8, Aalborg.

Li, Z., Zhu, Q., and Gold, C., 2005. Digital Terrain Modeling, ISBN 0-415-32462-9, CRC Press , Boca Raton.

Congalton, R.G., Green, K., 2009. Assessing the Accuracy of Remotely Sensed Data, 2nd ed. CRC Press, Boca Raton, U.S.A. ISBN 978-1-4200-5512-2.

Höhle, J., Höhle, M., 2013. Generation and assessment of urban land cover maps using high-resolution multispectral aerial cameras, *International Journal on Advances in Software*, vol. 6, no 3 & 4, year 2013, http://www.iariajournals.org/software/ 11 pages. (Article #1)

Höhle, J., 2017. Generating topographic map data from classification results. Remote Sens. (Basel) 9, 224.

Höhle, J., 2021. Automated mapping of buildings through classification of DSM-based ortho-images and cartographic enhancement, *International Journal of Applied Earth Observations and Geoinformation* 95 (2021), 102237, 11p.

ISPRS WG III/4. 2D Semantic Labeling Contest, 2014. 2D Semantic Label. - Vaihingen (isprs.org) (accessed on 27 February 2023)

Mikhail, 1976. Observations and Least Squares. IEP-A Dun-Donelley Publisher, ISBN 0-7002-2481-5, New York.

Potůčková, M., and P. Hofman, 2016. Comparison of quality measures for building outline detection. The Photogrammetric Record 31(154):193–209.

Rutzinger, M., Rottensteiner, F., Pfeifer, N., 2009. A comparison of evaluation techniques for building extraction from airborne laser scanning. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 2(1): 11-20.

CHAPTER 4. Applications

4.1 Introduction

This chapter covers some applications carried out by the authors. They reach from mapping of topographic objects (buildings, roads, parking lots, trees, grassland, bushes, hedgerows) to non-topographic objects (cars). The mapping of buildings in vector format has special attention. The enhancement of buildings in raster format will also be demonstrated by an example. R-programs may be used in "Moodle" where the generation of results can be followed step by step.

4.2 Methodology

The applied methodology is depicted by the flowchart in Figure 22.

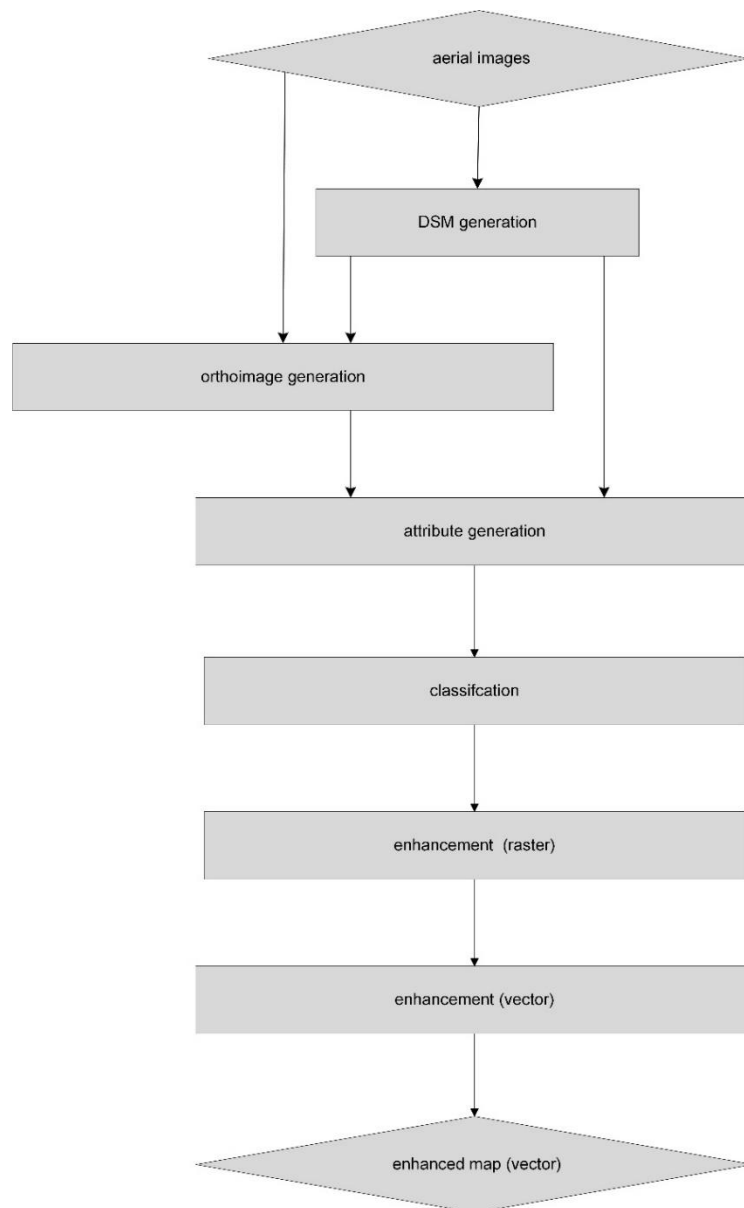Figure 22: Flowchart of the steps of the applied method for mapping of topographic objects.

Source materials are overlapping aerial images. By means of image matching a digital surface model (DSM) of high density is generated. Attributes which characterize topographic objects are derived from the orthoimage and the DSM. A classification of the DSM-based orthoimages generates land cover maps (LCM) of several classes including topographic objects (buildings, roads, and vegetation). Thereafter, an enhancement to raster and/or vector maps is carried out. This general methodology can be shortened if the necessary data are available from existing sources. In the following, the used source data, details of the processing, and the achieved accuracies of the examples are presented.

4.2.1 Mapping of buildings in vector format

Data used in the first two examples are DSM, false-colour orthoimage (DSM-based, with NIR- R-, and G-bands), and normalized DSM (nDSM). Furthermore, manually compiled land cover maps, are applied for the derivation of a classifier. These source data were available from (ISPRS WG III/4, 2014). Up to 17 features of the selected topographic objects were derived from these data.

A supervised classification automatically produces an LCM which is thereafter enhanced by the described method. The enhancement to vector data of a building outline is depicted in Figure 23.

a   b   c   d
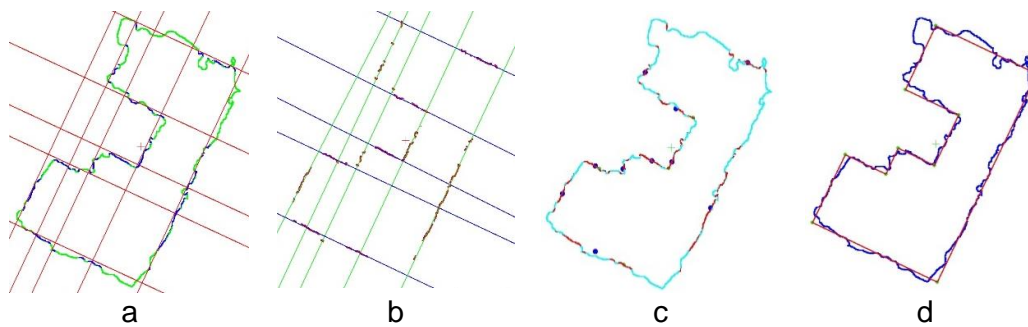
Figure 23: Processing of building outline (a) detected lines, (b) adjusted lines, (c) line sequence, (d) adjusted corner points and connecting lines.

The assessment of the cartographic quality and geometric accuracy is carried out by overlaying the vectors onto two references (ortho-image, manually compiled land cover map). The standard deviation of the coordinate errors (σ_x, σ_y) and the number of gross errors ($n_{gross\ error}$) are used as measures of the geometric accuracy.

Example 1

With the source data a land cover map of five classes has been produced using a DT classifier. 500 pixels per class have been used for training of the classifier. The modeling of the classes applied 17 attributes (NDVI, nDSM, INIR, IR, IG, AP). The attribute profile (AP) is derived from the three channels using three area thresholds (100, 1000, and 5000 pixels) and delivers 12 additional intensity values (DN). The thematic accuracy of buildings was PA=92% and UA=82%. The enhancement of the building outline used the program package 'buildenh' described in Appendix/Module 4/software tools.

The quality of the result can visually be judged by superimposing the derived vectors onto the two references (cf. Figures 24 and 25).

Figure 24: Enhanced vector data of building outlines superimposed onto the DSM-based orthoimage.
The green dots indicate undetected buildings. (Source: Höhle, 2021)

Figure 25: Enhanced building outlines (red lines) superimposed onto the manually compiled land
cover map (reference). The classes of the land cover map are coded by colours: "building" (blue),
"impervious surfaces" (white), "low vegetation" (cyan), "tree" (green), , "car" (yellow).
(Source: Höhle, 2021)

The outlines of the buildings are depicted in Figure 26. The plotted coordinates are in pixels of the orthoimage system.

Figure 26: Vector data derived from the DSM-based orthoimage. (Source: Höhle, 2021)

The achieved geometric accuracy is derived from a comparison of 130 (142) well-defined building corners. The average standard deviation is σ_{x,y_av} = 1.0 m (cf. Table 3). Two buildings were not detected.

Table 3: Geometric accuracy of derived corner coordinates.

Reference	Land cover map		Ortho-image	
Coordinate	x	y	x	y
σ [m]	1.1	0.9	1.2	1.0
n_{corner}	130		142	
$n_{gross\ error}$	2		2	

Example 2

Source data were DSM, orthoimage, nDSM, and reference map. All data cover area #1 of the ISPRS data set. Six classes ("building", "impervious surfaces", "low vegetation", "tree", "car", "clutter/background") were selected for the land cover map. The DT classifier is trained by means of

44

5.3 million pixels of the reference map (area #26) containing 'truth' values for all six classes. Only five attributes (I_{NIR}, NDVI, nDSM, $\sigma_{Z\ 5x5}$, $\sigma_{I\_NIR\_5x5}$) were used to model the classification. The achieved thematic accuracies of buildings were PA=78% and UA=88%. The processing of the building outline used again the program package "buildenh". The quality of the result may be evaluated by the vector plot and by overlaying the vectors onto the orthoimage (cf. Figures 27 and 28).

Figure 27: Vector data derived from a DSM-based orthoimage by the software package "buildenh". (Source: Höhle, 2021)

Figure 28: Generated vector data of buildings superimposed onto the DSM-based ortho-image. The green dot marks an undetected building. (Source: Höhle, 2021)
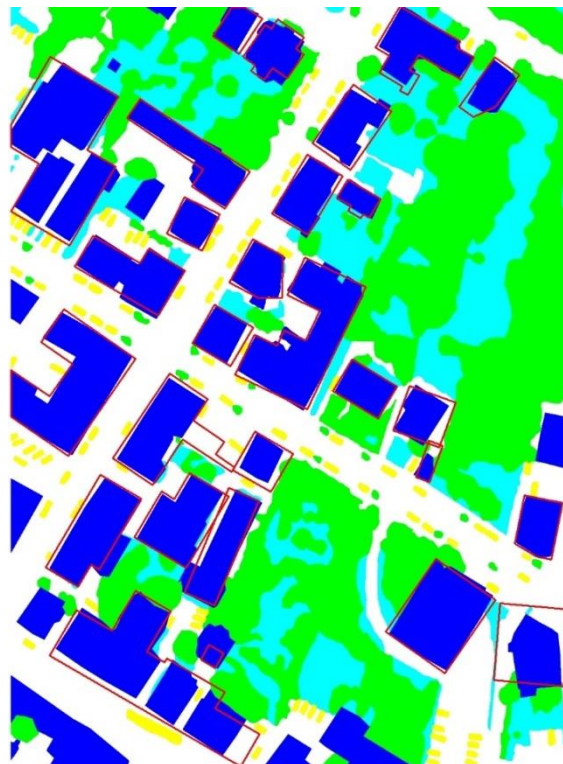
The achieved geometric accuracy is derived from 134 (154) building corners (cf. Table 4). The average standard deviation is $\sigma_{x,y\_av} = 0.9$ m. One building was not detected.

Table 4: Geometric accuracy of derived corner coordinates.

Reference	Land cover map		Ortho-image	
Coordinate	x	y	x	y
σ [m]	1.0	0.9	0.9	0.8
n_{corner}	134		154	
$n_{gross\ error}$	1		1	

More details on the last two examples are published in (Höhle, 2020).

Example 3

The source data of example 3 are overlapping aerial images of four bands (NIR, R, G, B) and their orientation data. A digital surface model (grid size 0.25 m), a digital terrain model (DTM), a normalized digital surface model (nDSM), and a false colour orthoimage (based on a DTM) were derived from the images. Six classes (″building″, ″road & parking lot″, ″wall & car port″, ″tree″, ″hedge & bush″, ″grass″) and a DT classifier were selected for the generation of the land cover map. The training of the classifier used 17449 pixels of the orthoimage containing 'truth' values for all six classes. Only two attributes (NDVI, nDSM) were used to model the class value. The thematic accuracies of buildings were PA=78% and UA=99%. The enhancement of the building outline used a preliminary version of the program package "buildenh". The achieved geometric accuracy is derived from 62 building corners (cf. Table 5). The average standard deviation is $\sigma_{x,y\_av} = 0.85$ m.

Table 5: Geometric accuracy of derived corner coordinates.

Reference	Orthoimage	
Coordinate	x	y
σ [m]	0.8	0.9
n_{corner}	62	

The obtained accuracy is related to the orthoimage, which in this example is based on a DTM. Figures 29 and 30 depict the building vectors overlaid onto the ortho-image and the vector map.
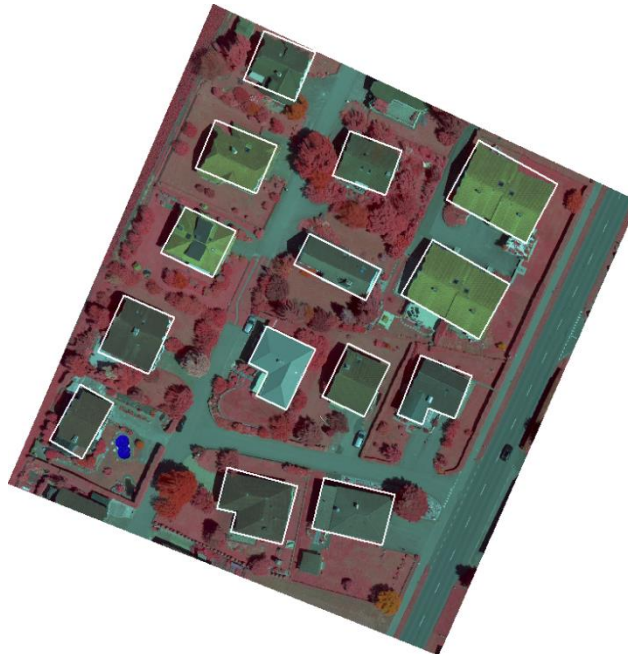

Figure 29: Generated vector data of buildings superimposed onto the orthoimage.
(Source: Höhle, 2014)
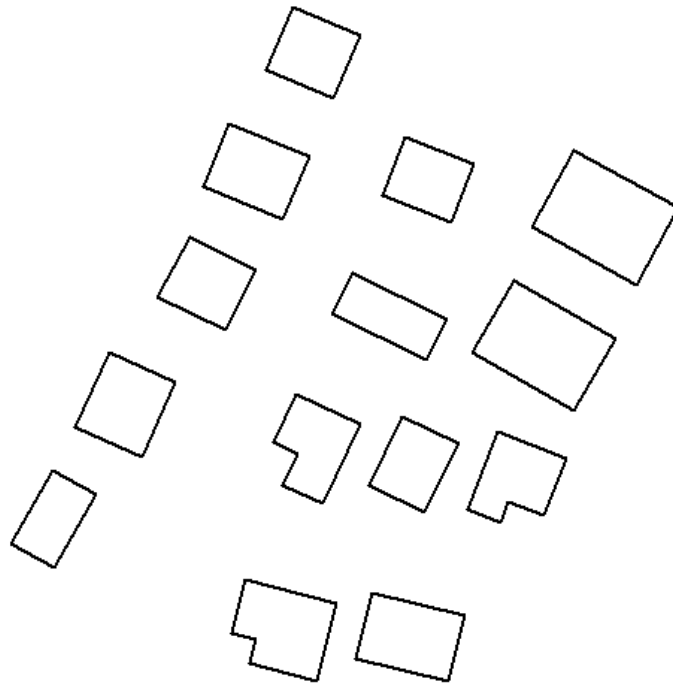
Figure 30: Enhanced vector data of class 'building'. (Source: Höhle, 2014)

More details regarding Example 3 are contained in (Höhle, 2014).

4.2.2 Related work

Automated generation of buildings and other man-made objects in vector format can be carried out by different methods. The presented approach uses aerial images as source data and generates first DSM-based orthoimages. The classification of the orthoimages results in a land cover map from which topographic objects are extracted and enhanced. The land cover maps are produced by a method from machine learning (Decision Tree). The generation of vector data applies the Hough transform and least squares adjustment for the detection of lines. The sequence of the detected lines is determined by ordering angles between object and line centres and/or by tracing the outline of connected pixels and searching for plotted points representing detected line segments. Successive and adjusted lines are then intersected. In case of buildings, the coordinates of the corners are improved using least-squares adjustment. The generated outline of the building will have perfect orthogonal and parallel lines.

Other research work in the category "Images as input data" has been done by Tarantino and Figorito (2011). True colour images were used to produce DSM, nDSM, and orthoimage. Several mask-layer data were derived and processed to building outlines by means of various image processing and GIS tools. The extracted outlines were simplified using the Douglas-Peucker algorithm (Douglas and Peucker, 1973). The assessment of 51 buildings of a residential area revealed a detection percentage of 92%. The planimetric accuracy was not determined. In (Bulatov et al., 2012), the outlines of the buildings have been extracted from a derived DSM using a minimum bounding rectangle. A planimetric accuracy of $RMSE_d = 1.2$ m has been achieved. The planimetric distances of vertices to lines (d) bigger than 3m were eliminated when computing the geometric accuracy. The approach by Wang (2016) also used a dense point cloud derived by image matching. Buildings are extracted by thresholding the gradients of elevations. The edge pixels of the roof are then traced, and closed polygons are generated. By a split-and-merge process, line segments are obtained, and straight lines

are fitted to them. Successive lines are intersected, and roof corners are obtained. 99% of detached buildings were detected. Geometric accuracies were not quoted.

High-resolution panchromatic satellite images were also applied to generate building outlines, e.g., in (Partovi et al., 2017). From overlapping WorldView-2 images, a low accurate DSM is derived and used to generate a building mask. The mask is further refined by classifying the geometrical features of the images. By means of a tracing algorithm the boundary points are extracted, and a set of line segments is then fitted to the extracted points by least-squares adjustment. The obtained line segments are intersected and connected to building outlines. Their geometric accuracy was assessed by the PoLiS method (Avbelj, 2015). An average value of distances to manually derived reference data was about two pixels corresponding to one meter.

Other investigations start from Airborne Laser Scanning (ALS) and identify, trace, and regularize the outlines of buildings. In (Awrangjeb, 2016), e.g., a Delaunay triangulation is applied to an ALS point cloud. A boundary line consisting of small segments can then be identified. The lines segments are smoothed, and corners are derived. The method was applied to six different areas. The achieved accuracy is quoted with $RMSE_d$=0.65m.

The vectorizing of a classification result has been carried out by other methods. Tasar et al. (2018) placed a fine mesh on top of the classified map. Labelled triangle meshes are then optimized by means of an objective function, which balances the closeness to the classification map, the rectangularity of the building edges, and the mesh complexity (cf. Figure 31). The approach yields an 80% overlap of the processed buildings to their reference areas. Small building edges can be detected and mapped. The generated building edges, however, are not perfectly straight, orthogonal, and parallel. Measures of geometric accuracy are not quoted.

(a) (b)

(c) (d)
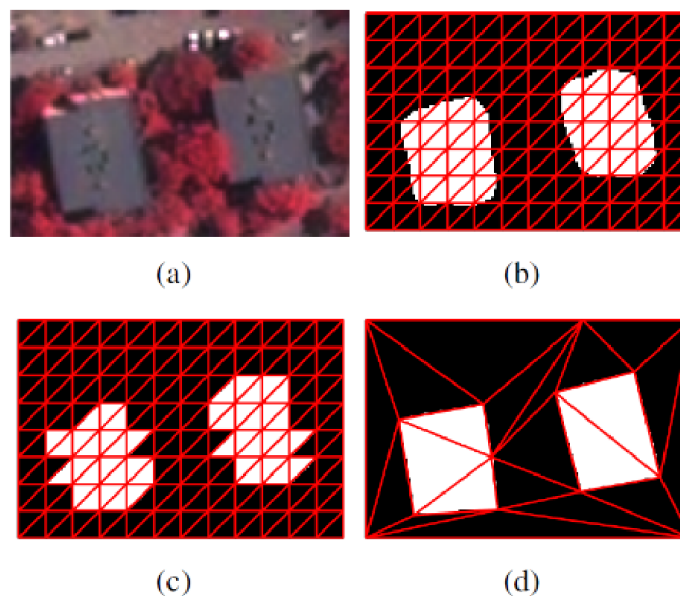
Figure 31: Input image and example labelled meshes. (a) Input image, (b) Initial fine lattice, (c) Initial and (d) Optimized labelled triangle meshes.
The triangles labelled as building are indicated by white colour. (Source: Tasar et al., 2018)

An algorithm simplifying complex lines, which is implemented as functions in the R package 'alphahull', is described in (Pateiro-López & Rodríguez-Casal, 2009). Other generalization algorithms are published by Reumann and Witkam (1974), and Visvalingam and Whyatt, (1992).

4.2.3   Mapping of other topographic objects

Experiences with other topographic objects in vector format are limited. Walls and hedges may also be represented by orthogonal and parallel lines. Objects with irregular outlines (roads, parking lots, vegetated areas) are processed as described in Höhle (2017). The map of the class ″road & parking lot″ is depicted in Figure 32. The segments are relatively small. Basically, it is difficult to detect and to extract small lines and to define their sequence in the polygon automatically. A simultaneous graphical output and some interactions will support error-free solutions.
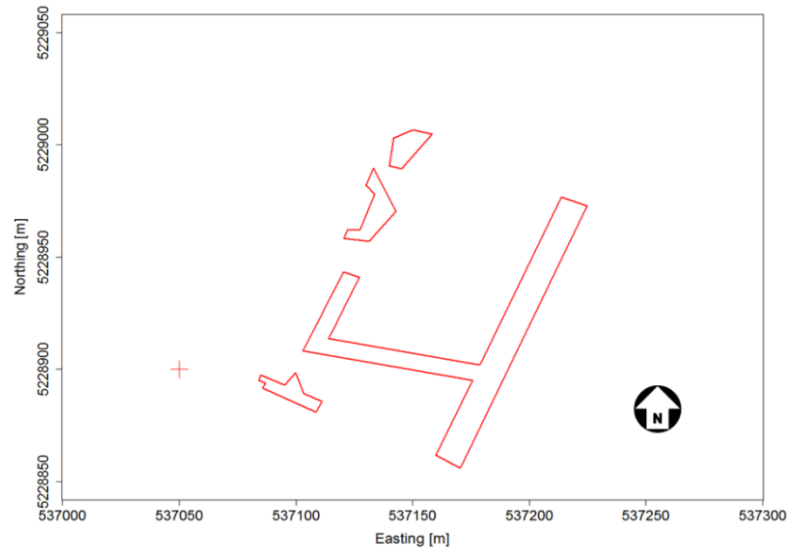
Figure 32: Enhanced vector map of class ″road & parking lot″. (Source: Höhle, 2017).

4.2.4   Enhancement to raster format

The generated land cover map can also be processed into an improved raster map. The topographical objects must be enhanced individually and are then joined again. Figure 33 depicts an example. The used data are the same as in Example 3.

Land cover map   Enhanced class 'building'   Enhanced raster map
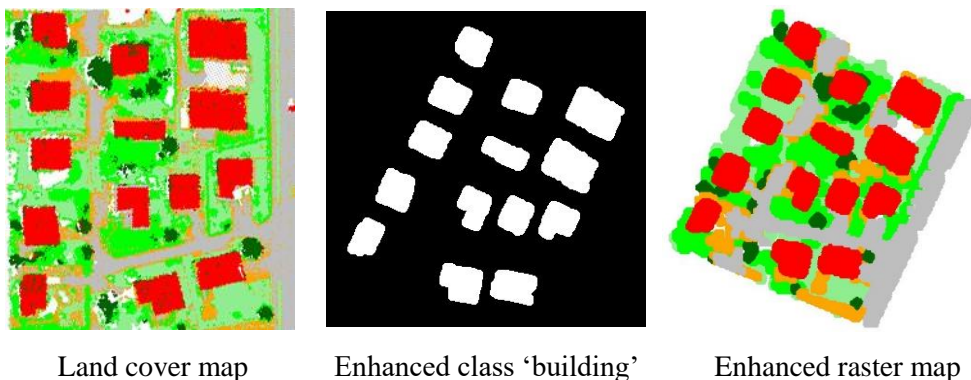
Figure 33: Enhancement of the classification result to a topographic map in raster format.
(Source: Höhle, 2014)

The homogeneity of the areas is improved, and small areas are removed. The topographic objects are labelled and supplemented with attributes (coordinates of the centre, area, and height above ground). Details on the methodology used are published in (Höhle, 2014).

4.2.5   Detection of hedgerows and small patches of trees

In this section, we will see an application in the detection of hedgerows and small woody patches using techniques which we have learned in Chapter 1 (Attribute profiles) and Chapter 2 (Classification). First, we will describe about the problem, and then we layout the solution.

Here the task is to detect small woody features from the high-resolution images. The detection of small woody patches and hedgerows are very important to the mapping agencies to monitor the changes as well as to maintain the ecosystem. The hedgerows have been pivotal part of the countryside, which provides food and shelter for many species, and in tacking the climate changes, providing numerous ecological and socio-cultural functions related to soil and water conservation, climate protection and adaptation, biological diversity, and cultural identity. Thus, the detection of hedgerows (small woody features) helps to agencies to design the best practices to maintain the healthy hedgerows. Hedgerows and tree groups are linked to landscape richness and fragmentation of habitats with a direct potential for restoration while also contributing to hazards protection and green infrastructure, amongst others. The specific ecological importance of SWF underpins the need for reliable, detailed geospatial information on the occurrence and distribution of linear landscape features. SWF are an elementary part of a landscape's green infrastructure and are therefore addressed through a range of policies and directives like EU's 2020 Biodiversity Strategy. Extracting these objects over such a large area (almost 6 million km^2) from VHR imagery brings numerous challenges: large amount of data (greater than 120 TB), large number of individual image scenes (greater than 38,000), diversity of the European landscapes, and need to process these data in a timely manner whilst ensuring a satisfactory degree of accuracy. Existing tools to automatically map the SWF partially overcome the above challenges but does not have the capability to tackle Big GeoData.

With these big data and designing effective feature extraction techniques important. The feature extraction technique also needs to be scale to large scale data, the conventional spatial feature extraction might be scalable to big data. The attribute profiles which we have seen in the Chapter 1 is effective in handling big data and can be parallelizable. Attribute profiles with the combination of random forest classifier is best for this problem.

4.2.6   Approach

As mentioned earlier we use attribute profiles and Random Forest. In attribute profiles, we use the variant known as differential attribute profile. The overall scheme of the approach is shown in Figure 34.
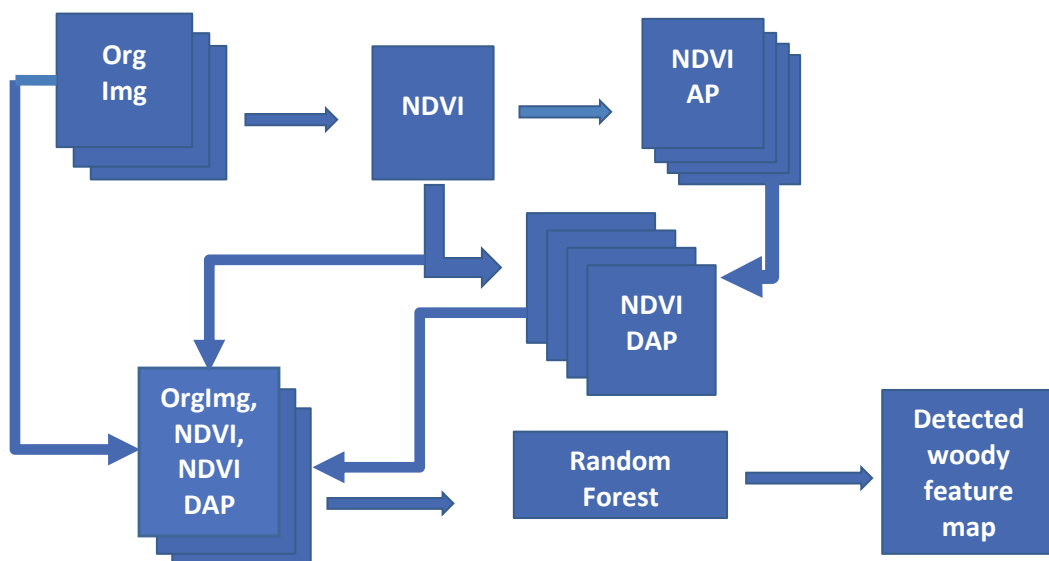

Figure 34: Overall schematic scheme of the detection of the small woody features using attribute profiles and random forest classifier.

Differential Attribute Profiles (DAP):

The differential attribute profiles are use to remove the objects which are larger than the given threshold. DAP is computed as follows:

1. Compute NDVI from the 4-channel satellite image
2. Compute attribute profiles on the NDVI with Area attributes with different thresholds. The thresholds are: 500, 750, 1000, 2500, 5000, 7500
3. To compute the DAP, find the successive difference between the attribute profiles computed in step 1

The example of DAP is shown in Figure 35, where it is visible that large part of the objects are removed. The above thresholds are well tuned for the detection of the small woody features, thus it may not be suitable for other applications.
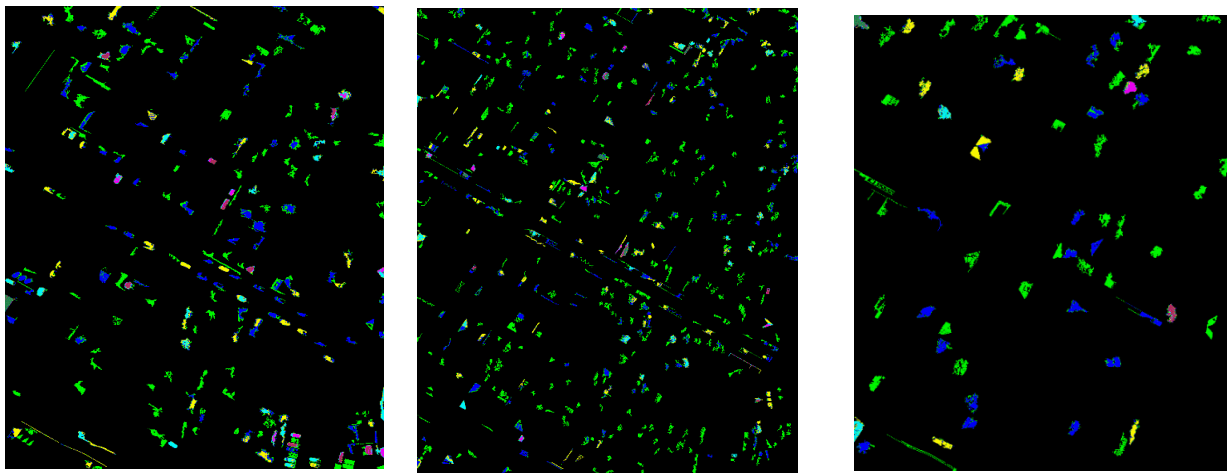
Figure 35: Differential attribute profiles (DAP) computed by successive difference of attribute profiles.

The DAP can be applied on original image, but we did not use it because it might not be effective for scalability to big data, as we need to compute attribute profiles on each channel of the image. So here use only NDVI. Apart from this NDVI can also well separate the vegetation regions from non-vegetation regions which helps reduce the variability in detection of the woody features.

For the detection of the small woody features, along with DAP computed on NDVI, we use original image and NDVI as the features for the random forest classification. The random forest classifier tries to distinguish between the woody features and non-woody features. In order to train the random forest classifier, we prepare the training samples using existing ground truth map. The training samples only contain the foreground objects, i.e, small woody features. For the background classes, the samples are randomly selected from the images and pruned by computing the distance to the mean of small woody features samples and we remove the samples which are closer the mean of the small woody features. We apply this pruning for two or three iterations, and then we train the random forest classifier with 100 decision trees.

Figure 36 shows the original image and resulting one of the differential attribute profile (DAP). One can see that large objects are removed, and it helps the random forest to effectively detect the small woody features. The small non-vegetation objects can be easily classified as background due to the NDVI map.
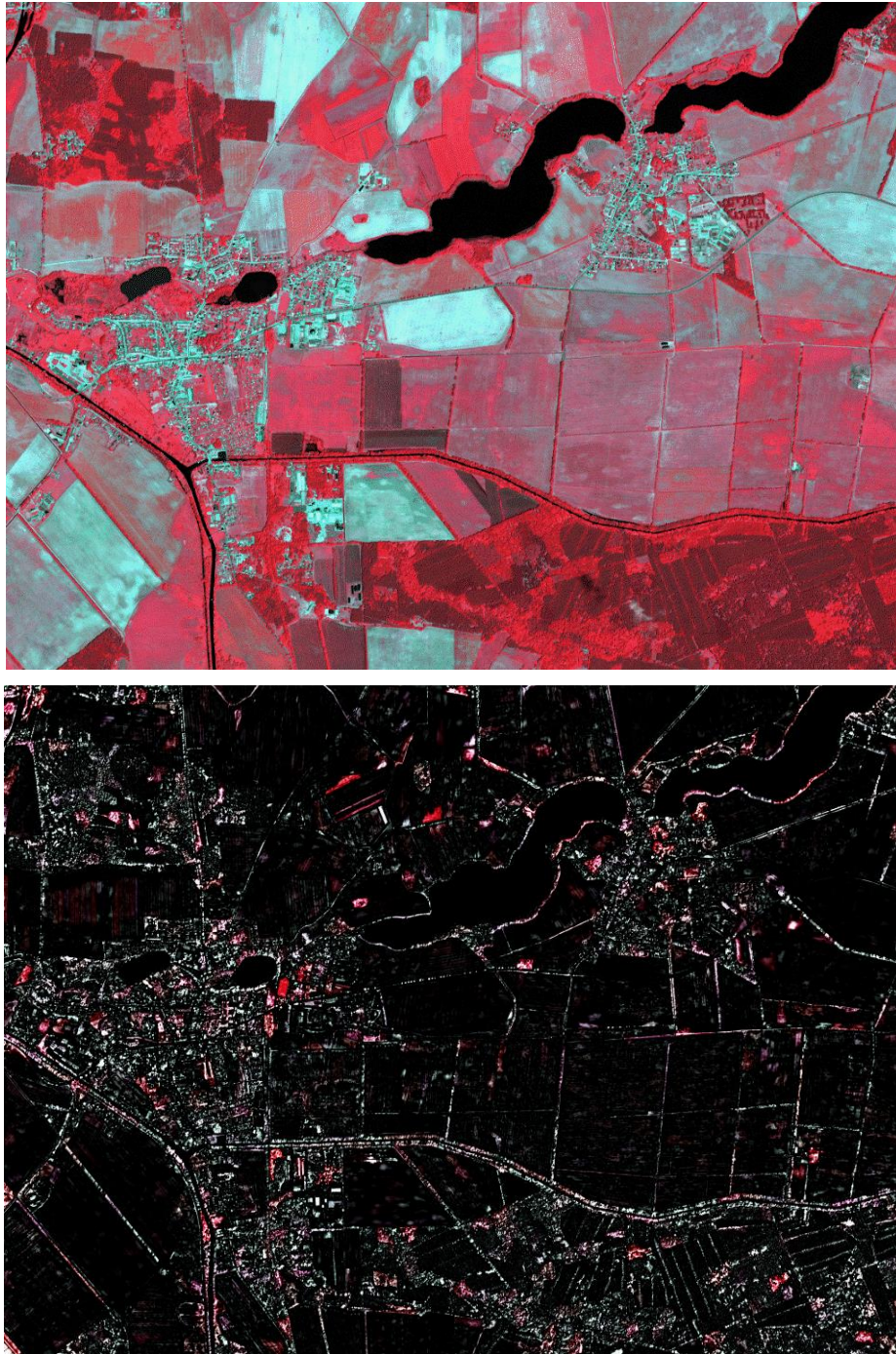
Figure 36: Original image (top) and the differential attribute profile (DAP) (bottom).

In Figure 37, on the left shows the original image and, on the right, shows the detected woody feature map (SWF map). One can notice, how well the combination of the differential attribute profiles and random forest classifier can effectively detect the small woody features.
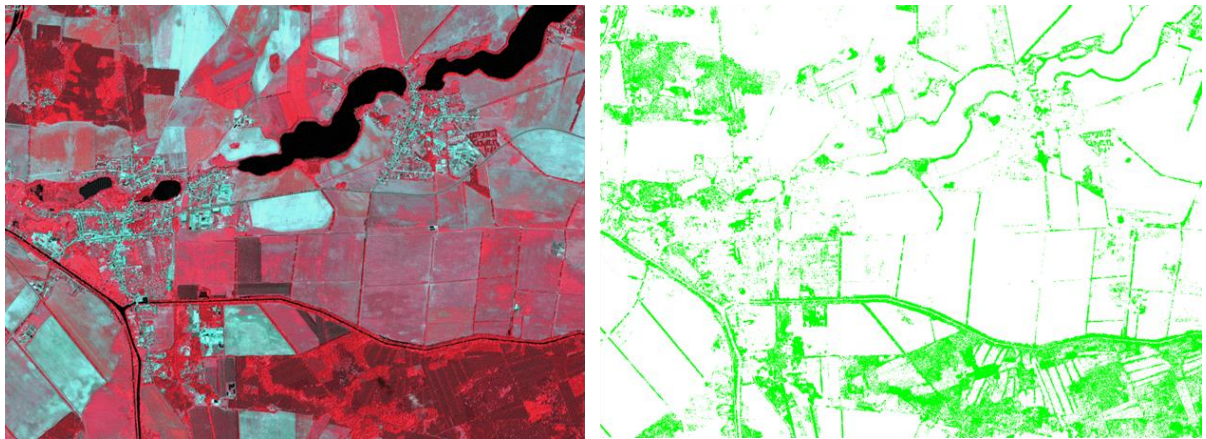
Figure 37: Source image (left) and small woody features (SWF) map (right) by differential attribute profiles and random forest.

The example is part of a European project covering 6 million km^2 (Merciol et al., 2019).

For the use of real-world applications, the detected woody feature map undergoes well known post-processing to increase the reliability of the map.

References

Avbelj, J., R. Muller, and R. Bamler, 2015. A metric for polygon comparison and building extraction evaluation. IEEE Geoscience and Remote Sensing Letters, 12(1):170-174.

Awarangjeb, M., 2016. Using point cloud data to identify, trace and regularize the outlines of buildings, International Journal of Remote Sensing, 37 (3): 551-579.

Bulatov, D., F. Rottensteiner, K. Schulz, 2012. Context-based urban terrain reconstruction from images and videos. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 1(3):185-190.

Douglas, D.H. and T.K. Peucker, 1973. Algorithms for the reduction of the number of points required to represent a digitized line or it caricature. The Canadian Cartographer, 10(2):112-122.

Gerke, M., 2014. Normalized DSM.
https://www.researchgate.net/profile/Markus_Gerke/publications
(accessed on 27 February 2023)

Höhle, J., 2014. Generation of 2D land cover maps for urban areas using decision tree classification. ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci. 2014, II-7, 15–21.

Höhle, J., 2017. Generating topographic map data from classification results,
Remote Sensing, 2017, 9, 224, doi:10.3390/rs9030224

Höhle, J., 2020. Automated mapping of buildings through classification of DSM-based ortho-images and cartographic enhancement (International journal of applied earth observation and geoinformation, open source, CC BY-NC-ND)
https://doi.org/10.1016/j.jag.2020.102237

Höhle, J., 2023. 'buildenh' - a collection of R-scripts for the cartographic enhancement of buildings generated by classification of remote sensing imagery (unpublished).

ISPRS WG III/4., 2014. 2D Semantic Labeling Contest.
2D Semantic Label. - Vaihingen (isprs.org) (accessed on 27 February 2023)

Merciol, F., Faucqueur, L., Damodaran, B.B., Rémy, P.-Y., Desclée, B. Dazin, F., Lefèvre, S., Masse, A., and Sannier, C., 2019. GEOBIA at the Terapixel Scale: Toward Efficient Mapping of SmallWoody Features from Heterogeneous VHR Scenes, ISPRS Int. J. Geo-Inf. 2019, 8, 46; doi:10.3390/ijgi8010046, 16 p.

Partovi, T., R. Bahmanyar, R., Krauss, T., and P. Reinartz, 2017. Building outline extraction using a heuristic approach based on generalization of line segments. IEEE Journal of Selected Topics in Applied Earth Observation and Remote Sensing, 10(3):933-947.

Pateiro-López, B., Rodríguez-Casal, A., 2019. Generating the Convex Hull of a Sample: The R package 'alphahull', R documentation.

Reumann, K. and Witkam, A. P. M., 1974. Optimizing curve segmentation in computer graphics. Proceedings of the International Computing Symposium.

Tarantino, E., Figorito, B., 2011. Extracting buildings from true color stereo aerial images using a decision-making strategy, Remote Sensing 2011(3):1553-1567.

Tasar, O., Maggiori, E., Alliez, P., and Tarabalka, Y., 2018. Polygonization of binary classification maps using mesh approximation with right angle regularity. IGARSS 2018 – 2018 IEEE International Geoscience and Remote Sensing, pp. 6404-6407.

Visvalingam, M. and Whyatt, J. D., 1992. Line generalisation by repeated elimination of the smallest area. Cartographic Journal, vol. 30, no. 1, pp. 46–51.

Wang, Y., 2016. Automatic extraction of building outline from high resolution aerial imagery. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XLI-B3, 419-423.

Appendix

1. Introduction

The following material is stored in the learning system "Moodle", it is structured as the Modules inside "Moodle". In this appendix, only the titles of the 'Literature to read' are named together with the number of pages; the bibliographic data are contained in the references of each chapter. Quizzes and assignments are presented by full text, also the solutions and answers. The software tools and data are described by text, the scripts can be accessed by a link to an Internet address. The material is based on the course that was hold in 2019 (EduServ17). The authors thank professor S. Lèfevre for providing his teaching material named as (Lefèvre, 2019). New software and new data were added in 2023.

2. Contents of Modules

Module 1

Literature to read
   This book (93 p.)
   Introduction to the course topic and course material (94 slides)

Supplementing material
   From classification results to topographic maps (lecture, 17 minutes)
   Enhancement to raster map (video, 1 minute)
   Enhancement to vector map (video, 1 minute)

Software tools
   Instruction for installations of R and RStudio
   Instruction for using the provided R-programs

Literature recommended to read
   #1: Automated mapping of buildings, 7 p.
   #2: DEM generation using a digital large format frame camera, 7 p.
   #3: A survey of image classification methods and techniques for improving classification performance, 47 p.

Supplementing literature
   Generation of Land Cover Maps Using High-Resolution Multispectral Aerial Cameras, 6 p.
   Dynamic range (11 sections)

Quiz

1) Accuracy of DEM (σ_h) derived by aerial images taken from an elevation of 10000 m above ground is: 0.5 or 2 or 5 m
2) The photogrammetric generation of a DEM by means of images with GSD=5cm. What is the DEMs density? 20 or 40 or 400 points/m^2
3) The dynamic range in a digital aerial camera is about 77 decibels (dB). What is the maximum digital number which may occur? 8192 or 16384 or 65536.

4) The status of the topographic maps in the world in scales 1:25000 and bigger is: 5% or 30% or 98% of the land area?
5) If the available orthoimage is based on a digital terrain model (DTM) instead of a digital surface model (DSM), how big is the displacement of a 10 m high house in the orthoimage? Use the example of the ISPRS data set (DMC camera, GSD=9 cm) and assume that a third of the aerial image is used for orthorectification. Select one: 1m or 3m or 8m.

Assignment

1) Name the characteristics of the different types of orthoimages. [Ref.: Article #1:"Automated mapping of buildings"]
2) What GSD should images have to derive a DEM with σ_h=0.1m. Use the new Leica DMC III camera (c=92mm, pixel size (pel)=3.9 μm, 25728 pixels x 14592 pixels) as example. [Ref.: Article #2: "DEM generation using a digital large-format frame camera"]
3) Explain the generation of colour images using pan-sharpening.
4) Some aerial cameras use the RGB Bayer pattern to generate colour images, e.g. RCD 30 and UltraCam Condor. What is the ratio of pixel sizes (GSD) between the spectral bands in these two cameras and what are the consequences for classification work?
5) What are the characteristics of buildings, roads and trees and which data (attributes) may help to detect them? [Ref.: Article #4: "Generation of Land Cover Maps Using High-Resolution Multispectral Aerial Cameras"]
6) Describe the main steps of an automated process for topographic mapping from remotely sensed imagery. For each of these steps, identify the main challenges. [Ref.: Article #3: "A survey of image classification methods and techniques for improving classification performance"]
7) Explain how each pixel can be described (i.e., by which features) to assign a topographic / land cover class using a supervised classifier. [Ref.: Article #3: "A survey of image classification methods and techniques for improving classification performance"]

Solutions to Quiz

Ad 1.: Correct answer: 2m (cf. article #2, Abstract)

σ_h=0.2%$_0$ h

σ_h=standard deviation of elevation

h=flying altitude,

σ_h=0.0002·10000 m=2 m

Remark: Using advanced cameras, higher overlaps of the images, and new methods for image matching, much better results can be achieved nowadays. In (Ressl 2016), e.g., a standard deviation of the elevation σ_h= 7cm or 0.7 GSD has been achieved for DTM accuracy. This corresponds to 0.035 %o of the flying altitude.

Ad 2.: Correct answer: 400 points/m^2

Density=1/GSD^2

GSD=Ground Sampling Distance

Density=1/0.05^2=400 points/m^2

Ad 3.: Correct answer: DN_{max}=8192, DN_{min}=1

(cf. Reference Wikipedia)

$DR=20 \cdot \log_{10}(2^Q/1) = 6.02 \cdot Q$ (dB)

$Q=77/6.02=12.8$ bit≈13 bit

2^{13}=8192

Remark: Newer digital aerial cameras have a DR of 84 dB where the maximum digital number (DNmax) is16384=2^{14}. Many more details can then be detected in areas with shadows.

Ad 4.: Correct answer: 30% of world land area.

This means that a lot of work must be done. Efficient methods are required. They are even more demanded for updating of the maps/databases. According to the study of (Konecny et al., 2015) the current update rate in this map category is 10-30 years only. (Ref: "Introduction to course topic and course material", slide #11).

Ad 5.:

Meaning of notations in question 5 are depicted in the following figure, showing the displacement of a house (grey area) in a DTM-based ortho-image.

Figure 1. Displacement of a house (grey area) in a DTM-based ortho-image.
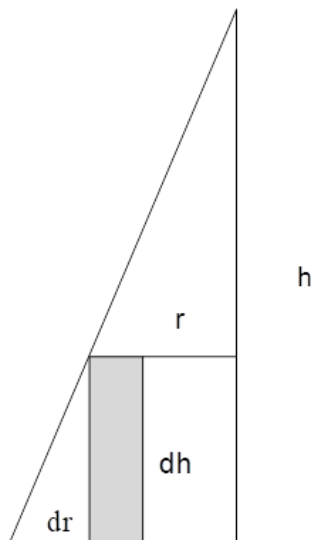
dr = r·dh/(h-dh)

DMC: camera   constant   (c)=120mm,   pixel   size (pel)=0.012mm,
image format: sp=7680 pixels, sq=13824 pixels
ISPRS data set (Vaihingen): GSD=9cm

h=c·GSD/pel
r=sqrt(sq^2 + sp^2)/2

orthorectification: r_red ≈ r/3

h=120·0.09/0.012=900m
dr=7907·10/(900-10)=8m
dr_red=8/3=2.7m (≈3m)

The correct answer is: The displacement (dr) of a dh=10 m high house is about 3m at the edge of the central part of the aerial image which has been used for the orthorectification.

Ad 1.

General characteristics: georeferenced (known transformation matrix), corrected for tilts and changes in elevations, uniform scale, pixel size of orthoimage, GSD of aerial images, time of photography, maximum image angles, different types (colour/CIR, DEM/DSM, single image/multi-image)

Individual characteristics:
DSM-based orthoimage: wiggly lines for roof edges, ghost images
DEM(=DTM)-based orthoimage: displacements at objects above ground, facades are visible, sharp edges at building outlines
True orthoimage: 3D building model with roof structure is required, multiple images for filling non-visible areas with image texture.

Ad 2.

After formulae and tests in reference (Höhle, 2009):

$$h = \frac{\sigma_h \cdot c \cdot b/h}{\sigma_{px'}}$$

h = flying height above average ground elevation
b/h = base/height ratio
σ_{px}' = matching accuracy
c = camera constant (calibrated focal length)

$m_b = c/h$ (Image scale number)

$GSD = pel \cdot m_b$ (Ground sampling distance)

pel = pixel size in image

with data of the DMCIII camera: c = 92mm, pel = 3.9µm=0.0039mm, s' = 14592 pixels (image size in direction of flight)
assumed data: p = 60% (standard forward overlap), σ_{px}'=0.6 pixel (as in reference #2)

$b/h = b'/c$
$b' = 0.4 \cdot 14592 \cdot 0.0039 = 22.76$ mm
$b'/c = 22.76/92 = 0.25$
$h = 0.1 \cdot 92 \cdot 0.25/(0.6 \cdot 0.0039) = 982.9$m
m_b = 982.9/0.092 = 10684
GSD = 0.0039·10684 = 42mm

GSD = 4.2 cm

Remarks: The value for σ_{px}' (0.6 pixel) was derived from a practical test with terrain points, the UltraCamD camera, and correlation techniques. It is recommended to determine at other conditions (correlation software, terrain type, camera) a 'practical' matching accuracy which can then be used for other DEM projects. The formulas in the reference may be applied. In recent literature (Ressl et al., 2016) a matching accuracy of σ_{px}'=0.2 pixel could be achieved by using higher overlaps and more advanced hardware and software. Note that the calculation of the flying altitude (h) requires the parameter σ_{px}' and b/h.

Ad 3.

High-resolution colour images are generated by combining a high-resolution panchromatic image with images of the spectral bands of lower resolution. The ratio between the pan- and the colour-resolution is at aerial cameras, e.g., 1:3.6 (UltraCamD) and 1:3.1 (DMCIII). Different techniques exist. At IHS pan sharpening, e.g., the low resolution RGB bands are first up sampled to the same resolution as the panchromatic band and then transformed into the I(ntensity), H(ue), S(aturation) colour space. The I-channel is substituted by the panchromatic channel and a back-transformation produces a high-resolution colour image with RGB bands (cf. Figure 2).

low-resolution ms data   high-resolution pan data   combined data
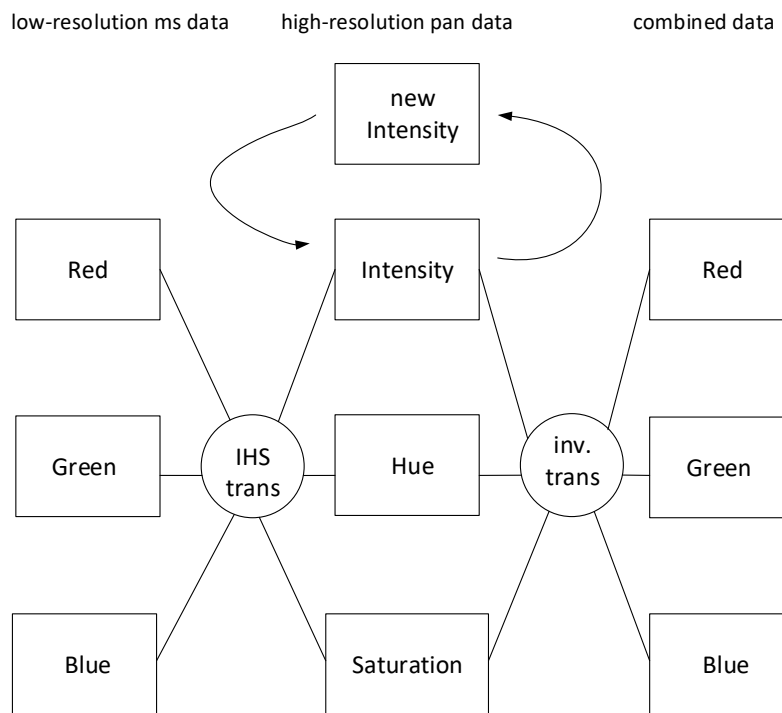
Figure 2. Principle of pan-sharpening (Source: Albertz, 1991)

Remarks: The RCD30 camera does not record a pan-chromatic band. The UltraCam Condor has a panchromatic band of lower resolution than its multi-spectral bands (R, G, B, NIR). Their colour images are produced by means of the Bayer pattern. Cameras with a Bayer pattern have colour filters in front of the sensor. The filters are arranged in a 2x2 matrix. The red and blue filters cover each one pixel of the 2x2 matrix, and two pixels are covered by green filters. Each pixel records therefore one colour only, the other two colours are interpolated from the neighbouring pixels. The resolution of the sensor is higher for the green colour by a factor of two. Images with the Bayer pattern may have some artefacts (wrong colours, Moiré pattern).

Ad 4.

The pixel size on ground (GSD) is:

$GSD = pel \cdot h/c$

pel = pixel size in image plane
c = camera constant
h = flying altitude above average terrain

UltraCam Condor:

The GSD ratio between the bands is RGB:PAN:NIR = 1:2.83:4.35 (cf. Table 1). The near infra-red (NIR) band has much lower resolution which may affect the quality of the false colour orthoimage and the accuracy of the NDVI. The production of true colour orthophotos can use the highest resolution. The panchromatic band is used for aerotriangulation and DSM production. High geometric accuracy is achieved for both tasks. The high base/height ratio (b'/c = 0.70) enables that.

Table 1: UltraCam Condor aerial camera

UltraCam Condor:	RGB [mm]	PAN [mm]	NIR [mm]
Pixel size (pel)	0.0046	0.0052	0.0046
Camera constant (c)	100	40	23
Ground Sampling Distance (GSD) at flying height (h) of 1000m	46	130	200

GSD ratios: RGB/PAN=1:2.83, RGB/NIR=1:4.35

RCD30:

The RCD30 camera produces natural colour and false-colour images by one camera head. The NIR band is imaged on a second CCD of the same pixel size (0.006mm). The RGB bands and the NIR band are co-registered and corrected for geometric and radiometric errors in post-processing. The RCD30 camera has not a panchromatic band.

The camera data and the GSD ratio between the spectral bands can be seen in the following table.

Table 2: RCD30 aerial camera

RCD30 camera (60 MP)	RGB [mm]	NIR [mm]
pel	0.006	0.006
c	50	50
GSD at h=1000m	120	120

There are no differences in the pixel size of the RGB and the NIR bands. The GSD ratio between the NIR band and the RGB bands is RGB:NIR=1:1.

Remark: The RCD30 is a medium format camera. The ground coverage is much less than that of the large format camera at the same GSD. More flight lines are required. A newer model has a smaller pixel (*pel* = 0.0052mm) and produces images of 80 megapixels (MP). Other lenses (*c* = 80mm or 150mm) can be used without a new calibration.

Ad 5.

The characteristics of some topographic objects are contained in the following table.

Table 3: Topographic objects characteristics

Topographic object	Characteristics	Data (attribute)
Building	height above ground planes uniform colour non-vegetated rectangular shape size	nDSM, slope, deviations of elevations normal vectors Digital Number (DN) of R, G, B, NIR bands NDVI (requires NIR band) major direction, parallel/orthogonal lines, shape, area, standard deviation in neighbourhood (e.g., 5x5 pixel)
Road	no height above ground not vegetated uniform colour	nDSM=0 NDVI DN of R, G, B, NIR bands (spectral signature)
Tree	height above ground non-uniform structure vegetated	nDSM spectral signature, texture NDVI (requires NIR band)

Ad 6.

(Source: Lefèvre, 2019)

A description of the overall framework was shown in the slides provided during the pre-course seminar, cf. learning material in "Introduction" (flowchart for automated topographic mapping). The proposed paper goes further and proposes the following (see Section 2):

(1) Selection of the remotely sensed data

Challenges: consider user's need, scale and characteristics of a study area, availability of images and their characteristics (e.g., atmospheric conditions, spatial resolution, etc.), cost and time constraints, analyst's experience in using the selected image.

(2) Selection of the classifier and the training samples

Challenges: define a suitable classification system and a sufficient number of training samples. For medium spatial resolutions: define adequate hierarchical levels for mapping, define discrete land-cover units, selecting representing training sites.

(3) Data pre-processing (image restoration, geometric rectification, or image registration, radiometric calibration, atmospheric correction, topographic correction)

Challenges: not detailed in the proposed paper (Section 2.3)

(4) Feature extraction and selection

Challenges: choose the appropriate features among a wide range of available variables, avoid using too many features/variables (Hughes phenomenon), ensure a good reference dataset to correctly assess the relevance of the features and to correctly model each class.

(5) Selection of a suitable classification method

Challenges: since different classification methods have their own merit, the choice of the most appropriate one is not easy (see Section 4.7).

(6) Post-classification performance

Challenges: deal with salt & pepper effect produced by pixelwise classification, address spectral confusion among land-cover classes, exploit ancillary data and/or expert knowledge (rules).

(7) Evaluation of classification performance

Challenges: select appropriate criteria to the problem in hand (most common criterion being classification accuracy, see discussion of criteria in Section 3).

Ad 7.

(Source: Lefèvre, 2019)

The paper proposes a list of features in Section 5. A pixel can be described based on various characteristics to build a feature vector used as input of a supervised classifier:

- spectral information: original pixel values (panchromatic, multispectral, hyperspectral) or derived indices such as NDVI, etc.
- spatial information: can be used in contextual or object-based classification; also, the basis of texture descriptors or attribute profiles
- temporal information: if multi-temporal images or satellite image time-series are available, each pixel is described by a time-series (and not only a spectral signature and spatial information acquired at a unique date)
- ancillary information: from a GIS, DEM, another data source, etc.

Module 2

(Source: Lefèvre, 2019)

Literature recommended to read
  #1: Introduction to spatial-spectral feature extraction (17 slides)
  #2: Morphological attribute profiles for the analysis of very high-resolution images (15 p.)
  #3: Recent developments from attribute profiles for remote sensing image classification (6p.)
  #4: Simple example of min- and max-tree (1p.)

Supplementing literature
  Feature profiles from attribute filtering for classification of remote sensing images (8p.)
  Local feature-based attribute profiles for optical remote sensing image classification (14p.)

Quiz

1) When computing an attribute profile over an image, a user sets an area threshold to 500. Select one:
   a. All pixels with value (intensity) higher or equal to 500 are removed.
   b. All pixels with value (intensity) lower than 500 are removed.
   c. All connected components with an area larger or equal to 500 are removed.
   d. All connected components with an area lower than 500 are removed.

2) Let us consider an attribute profile computed using the standard deviation attribute. If the image representation is in 16 bits, are the thresholds used for an 8-bit image sufficient or do they need to be updated?

   a. The thresholds for standard deviation depend on the image depth and thus need to be changed from 8 to 16 bits. The change consists in increasing (factor of two) the values of the thresholds.
   b. The thresholds for standard deviation are independent of the image depth and thus do not need to be changed from 8 to 16 bits.
   c. The thresholds for standard deviation depend on the image depth and thus need to be changed from 8 to 16 bits. The change consists in decreasing (factor of two) the values of the thresholds.

3) Instead of computing Attribute Profiles (AP), it is possible to compute Differential Attribute Profiles (DAP). (multiple possible answers)

   a. Object most representative scales can be extracted from the DAP.
   b. With DAP, we need a different number of thresholds than with AP.
   c. Computing a DAP from a min-tree or from a max-tree provides the same output for a given pixel.
   d. DAP are mostly sparse (lot of scales contain 0 values).
   e. DAP can lead to negative values (since they are computed as differences).

4) In order to compute attribute profiles from an image, we first need to set the thresholds for some given attributes (e.g., area, standard deviation, moment of inertia, etc.).

   a. The values to be selected for the thresholds should be the same for all attributes.
   b. The nodes of the max-tree always have attributes with increasing values from the leaves to the root.
   c. If more thresholds are used, it will be easier to discriminate between land cover classes.
   d. The chosen threshold values cannot be reused from one image to another.

Assignment 2

1) What are the differences between min-tree, max-tree, and tree of shapes (ToS)? Are there other tree structures that can be used to build attribute profiles?

2) What is the interest of using tree structures (min, max, ToS) to compute attribute profiles?

3) When is it relevant to choose respectively area, standard deviation, and moment of inertia to compute the attribute profile? Provide some examples of objects that might be extracted using each of these attributes.

4) What are the main steps when computing attribute profiles from an input image? For each of these steps discuss the main challenges.

5) What is the advantage of attribute profiles over morphological profiles? What is the advantage of differential attribute profiles over attribute profiles?

6) Now and for the next questions, consider the image below:
   (10 x 10 pixels, values between 0 and 5).

$$
\begin{array}{llllllllll}
0 & 1 & 1 & 2 & 2 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 2 & 2 & 0 & 0 & 0 & 1 & 1 \\
0 & 1 & 1 & 2 & 2 & 3 & 0 & 0 & 1 & 1 \\
0 & 0 & 1 & 2 & 2 & 3 & 4 & 1 & 1 & 1 \\
0 & 0 & 1 & 2 & 2 & 3 & 0 & 0 & 1 & 1 \\
0 & 0 & 1 & 2 & 2 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 5 & 5 & 0 & 4 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 5 & 4 & 4 & 4 & 0 \\
0 & 5 & 0 & 0 & 0 & 4 & 4 & 2 & 4 & 0 \\
0 & 0 & 0 & 0 & 0 & 4 & 2 & 2 & 4 & 4
\end{array}
$$

Build the corresponding max-tree.

7) Propose some area thresholds (not more than 3) that will be relevant to distinguish between the regions in the image when applied on the max-tree (thinning profile). Give the resulting filtered images obtained with these thresholds.

8) Build the corresponding min-tree.

9) Propose some area thresholds (not more than 3) that will be relevant to distinguish between the regions in the image when applied on the min-tree (thickening profile). Give the resulting filtered images obtained with these thresholds.

10) In this example, discuss the complementariness between the thinning and thickening profiles.

Assignment 2b (optional)

This assignment should be performed using the AP Tool (binary for Microsoft Windows).

1) For the given image "10m_cropped.tif", generate the attribute profiles using area attribute with max tree and min tree (use only one band in the image: (-b 0), option in the AP Tool). Observe and discuss the difference between the max tree and min tree. Use the following thresholds: Area Thresholds = [100, 500, 1000, 2000, 3500, 5000]

2) For the same image, compute attribute profiles with area attributes and discuss what happens when the thresholds of the area attribute are increasing.

Solution to Quiz

Ad 1.

When using attribute profiles, each connected component (or node in the tree) is compared to a threshold based on some specific features (called attribute). Keeping or removing a threshold will

result in modification of the grey levels of the input image. But the criterion used to do the selection is the attribute, i.e., area in this case. So, answers a. and b. are wrong since they are dealing with intensity. Furthermore, the goal of the filtering is to discard all components that do not fulfil a given criterion, here an area of 500. All components that do not have an area equal to (at least) 500 are removed. So, the correct answer is d.

Ad 2.

Here we consider computing attribute profiles with standard deviation as the selected attribute. In other words, for each connected component (or node in the tree), the standard deviation of pixels belonging to the component is measured. The standard deviation is measured based on pixel intensities. When changing the image representation from 8 to 16 bits, the image depth, i.e. the range of available/possible pixel intensities is doubled. This leads the observed standard deviations to be doubled too. The correct answer is then a. Let us note that for other attributes such as area or moment of inertia, the change of image depth will not affect the predefined thresholds.

Ad 3.

While AP are computed directly from the attribute filtering outputs, DAP are computed as difference between the successive responses of the attribute filtering process. Each value in a DAP is thus a difference between the greyscale returned using an attribute threshold and the greyscale returned with the next threshold. If a filtered component does not change between two successive filtering steps (i.e., with two successive attribute thresholds), the DAP will contain a 0 (to be compared with two successive identical values in the corresponding AP). In practice, this happens very often and thus DAP contain lots of null values. It is thus relevant to focus on the few non-null values of the DAP, that indicate the scales for which a pixel change (or more precisely the components it belongs to are filtered). Thus answers a. and d. are true.

There is no reason why DAP would need some different settings than the original AP it is built from (answer b. false). Furthermore, a max-tree (resp. min-tree) highlight bright (resp. dark) components, with a decreasing (resp. increasing) level from the leaves to the root. Structures of min- and max-tree are thus completely different, and there is no reason why they should output the same value for a given pixel (answer c. false). Finally, while DAP are computed as the difference between two successive values, their content is always positive (answer e. false): for a max-tree with decreasing values from leaves to root, the difference is computed as DAP(n)=AP(n-1)-AP(n); for a min-tree with increasing values, the DAP is defined as DAP(n)=AP(n)-AP(n-1).

Ad 4.

The proposed answers are independent, let us review them one-by-one. Of course, each attribute comes with its own support/domain, i.e., its own set of possible values. Areas cannot be compared to standard deviations nor to moments of inertia. Thus, it is mandatory to define specific thresholds for each attribute (answer a. is false).

In a max tree, the nodes (or connected components) are ordered based on their intensity (remember the leaves are the brightest components, while the root contains the whole image). This ordering is not related at all to the chosen attributes. Nevertheless, we can distinguish two kinds of attributes: increasing attributes which take low values for leaves and increasing values for their parents (or nesting components) up to the root of the tree. Area (or usually size-related attributes) is an example of increasing attribute: its value is always higher for a nesting component than the corresponding nested component. Beyond increasing attributes, we can use non-increasing attributes, i.e. attributes that does not ensure this increasing property. In other words, with non-

increasing attributes, we can have nesting components that do not have an attribute value higher than their nested components, and the minimal and maximal values of the attributes are not necessarily found in the leaves and root of the tree, respectively. Examples of these attributes are shape-related attributes (i.e., moment of inertia) or homogeneity/intensity-related attributes (i.e., standard deviation). So, answer b. is false too.

Considering a set of N thresholds will allow to build an attribute profile of size N, i.e., a feature vector of N values for each pixel. This feature vector corresponds to the maximal grey values of the selected components to which the pixel belongs, after each of the thresholds. In the context of machine learning and data classification, the length of the feature vectors corresponds to the number of dimensions of the associated data space. The more features we use, the more ability to discriminate between rather similar patterns we have. So, answer c. is true. Let us, however, note that too many features will lead to the curse of dimensionality (Hughes phenomenon) and it might then be necessary to perform a dimension reduction step afterwards (either through feature extraction or feature selection).

Setting appropriate features given a classification problem is not straightforward. As indicated previously, each attribute comes with its specific value range, and each image might require a specific tuning depending on its properties (spatial resolution, image depth, kinds of objects to be classified). Nevertheless, assuming two different images represent similar content (e.g., general land cover mapping problem), it is possible to reuse generic values for the attribute thresholds. For instance, the paper "Recent Developments from Attribute Profiles for Remote Sensing Image Classification" (article #3) used the same thresholds for the moment of inertia attribute, but different thresholds for the area attribute, in the classification of two different images. This is because the image resolution differs (thus size of objects also varies between the two images), the shape of observed objects remains image independent. So, answer d. is (mostly) false.

Answers to Assignment 2

Ad 1.

Min-tree highlights dark components of the image, and the levels (scales) of the tree range in an ascending order of grey levels. Max-tree is the dual representation: it highlights bright components of the image, and the levels of the tree range in a descending order. The Tree of Shapes (ToS) is a self-dual representation that allows to emphasize both bright and dark components into a unique representation. Leaves are the local extrema of the image (either bright or dark), and they are progressively nested into more central components, up to the whole image that corresponds to the root of the tree.

As indicated in the paper "Recent Developments from Attribute Profiles for Remote Sensing Image Classification" (article #3), it is also possible to rely on other tree structures such as partitioning trees to build attribute profiles. Such trees correspond to the different layers of a multiscale segmentation. In such models, every pixel of the image belongs to a leaf (pixels that are adjacent and similar may belong to the same leaf). So, the inner components of the tree are built only by merging neighbouring regions. The most popular models of morphological trees are alpha-tree, omega-tree, and binary partition tree.

Ad 2.

While it is possible to compute attribute profiles using attribute filters directly applied on the input image, representing the image through its tree structure allows the filtering to operate on components (nodes of the tree) and not pixels anymore. It leads to a great gain in terms of

efficiency. As explained in the previous answer, the choice of the tree structure (min, max, ToS) depends on the kind of objects we are interested in. In a given context, if only bright (resp. dark) objects are of interest, it is appropriate to use a max (resp. min) tree. If no assumption about the kind of extrema can be made (i.e., both dark and bright objects are relevant), then ToS is the most appropriate. Another option in the latter case is to use both a min- and a max-tree, but this leads to two attribute profiles instead of one with the ToS (in other words, the length of the feature vector is doubled w.r.t. using ToS).

Ad 3.

The attribute has to be chosen depending on the application. If the objects of interest could be discriminated based on their size, homogeneity (in terms of intensity), or shape, then it is relevant to use area, standard deviation, or moment of inertia respectively. Of course, it is possible to choose several attributes, thus, to build several attribute profiles (one per attribute) if combining these properties could help to distinguish better between the different classes. If no assumption about the object properties for a problem at hand can be made, then it is better to use multiple, complementary attributes. Please refer to the practical assignments which kind of objects can be extracted applying different attributes.

Ad 4.

As illustrated in "Recent Developments from Attribute Profiles for Remote Sensing Image Classification" (Section II), the main steps are: 1) construct the tree; 2) compute the relevant attributes for each node of the tree; 3) filter the tree based on some predefined attribute thresholds; 4) reconstruct the image from the filtered tree.

Among the challenges to be faced, we can cite:
1) building a tree from complex data, e.g., multi- or hyperspectral images, SAR images, LiDAR data (considering to build AP over derived features might also bring some additional information);
2) choosing the most appropriate attributes depending on the problem at hand;
3) selecting the right thresholds, possibly based on some prior knowledge or following a data-driven approach;
4) exploiting the filtered images, preferably by postprocessing them to reduce redundancy observed with AP (e.g., using derivative features such as DAP, performing feature extraction/selection, computing advanced measures such as feature profiles or local-feature attribute profiles).

Ad 5.

Morphological profiles have been introduced almost a decade before attribute profiles. They rely on successive morphological filtering, where the level/scale is defined by the size of the structuring element. Attribute profiles can be seen as an extension of morphological profiles which offer much more flexibility, since images are filtered not anymore based on the size of the structuring element, but on any attribute that can be computed over the connected components. Furthermore, thanks to the tree-based implementation, attribute profiles can be computed very efficiently conversely to morphological profiles. As far as differential attribute profiles are concerned, they provide a sparse representation that allows to identify the main scales of interest for every pixel (see the explanation provided in the answers at the feature extraction quiz).

Ad 6.

The max-tree is illustrated in the image below. Please note that the max-tree could have been simplified by removing redundant nodes (all nodes are kept here for the sake of understanding). See answer 7 for a simplified max-tree.

Figure 3: Max-tree

Ad 7.

The thinning profile computed by filtering the max-tree with 3 thresholds {2, 5, 25} on the area attribute is illustrated in the image below.

Figure 4: Thinning profile by filtering the max-tree with 3 thresholds {2, 5, 25} on the area attribute

Ad 8.

The min-tree is illustrated in the image below. Please note that the min-tree could have been simplified by removing redundant nodes (all nodes are kept here for the sake of understanding). See answer 9 for a simplified min-tree.
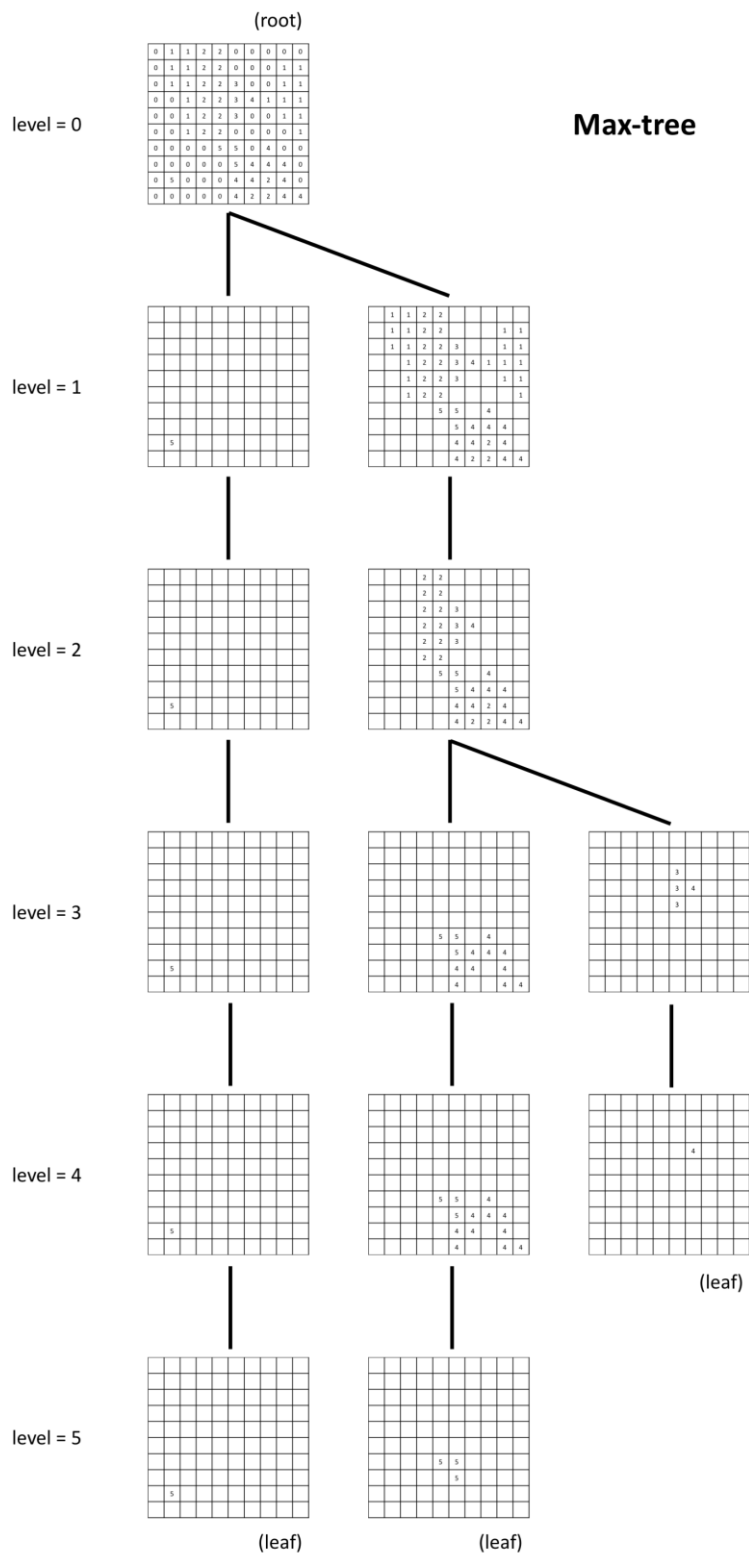

Figure 5: Min-tree

Ad 9.

The thickening profile computed by filtering the min-tree with 3 thresholds {15, 30, 60} on the area attribute is illustrated in the image below.
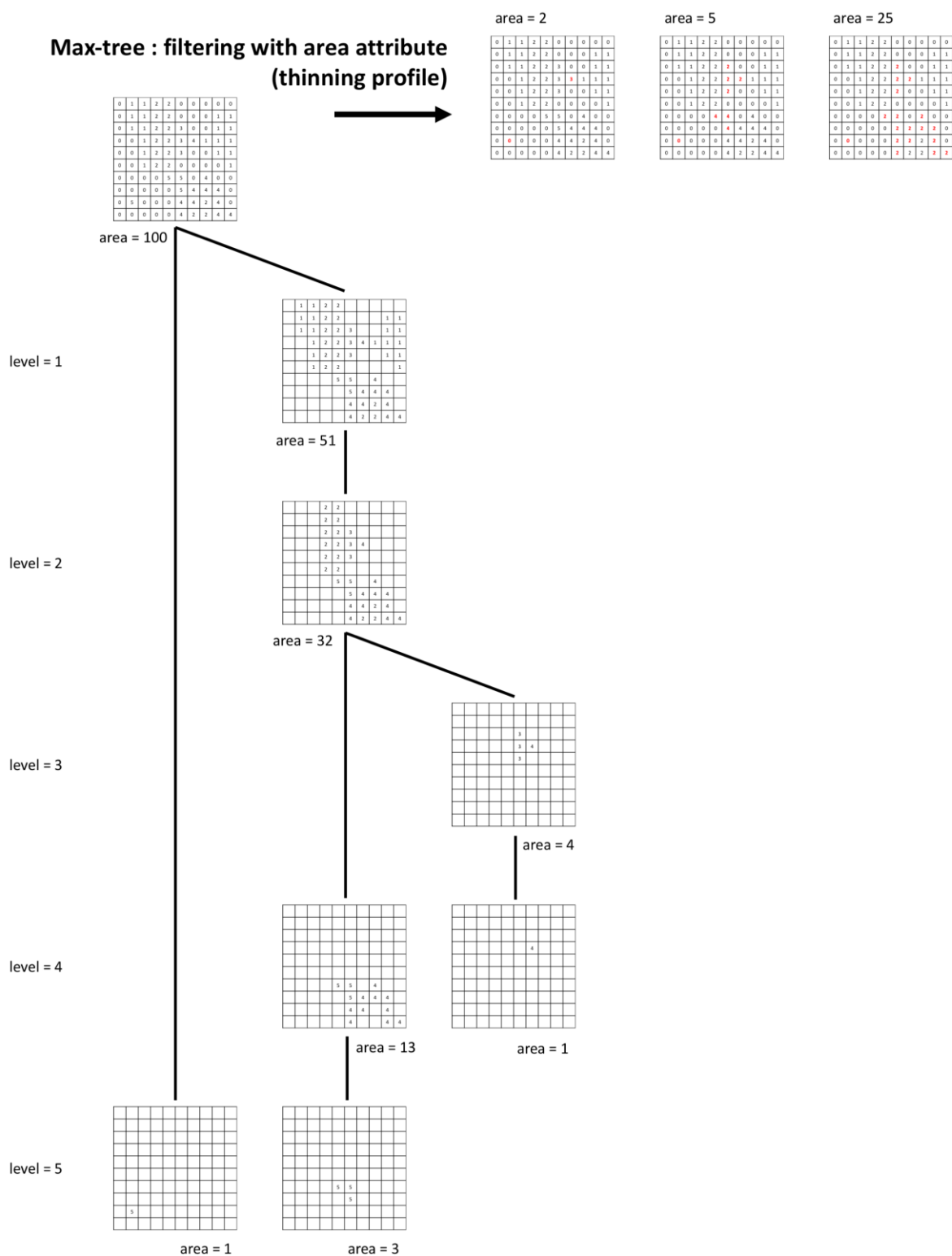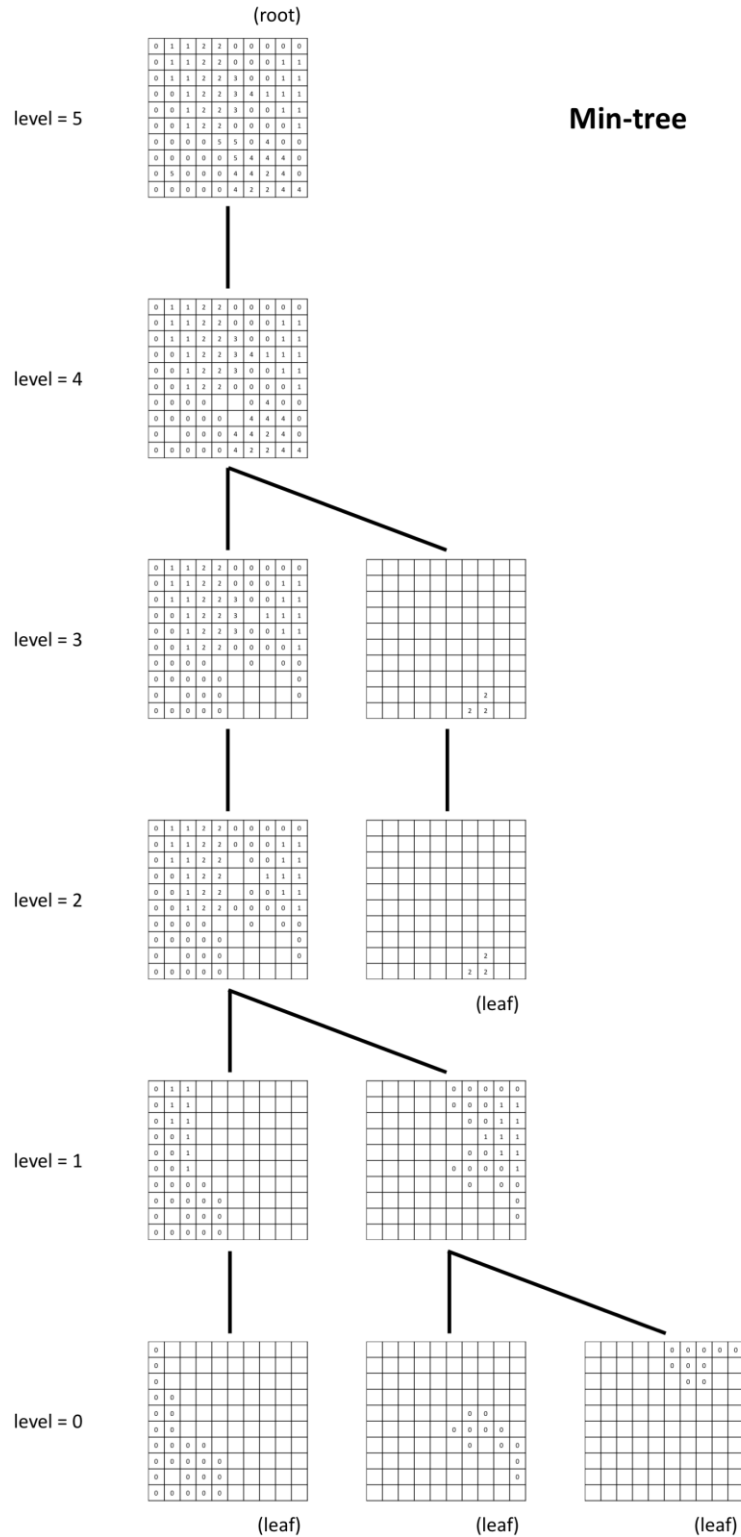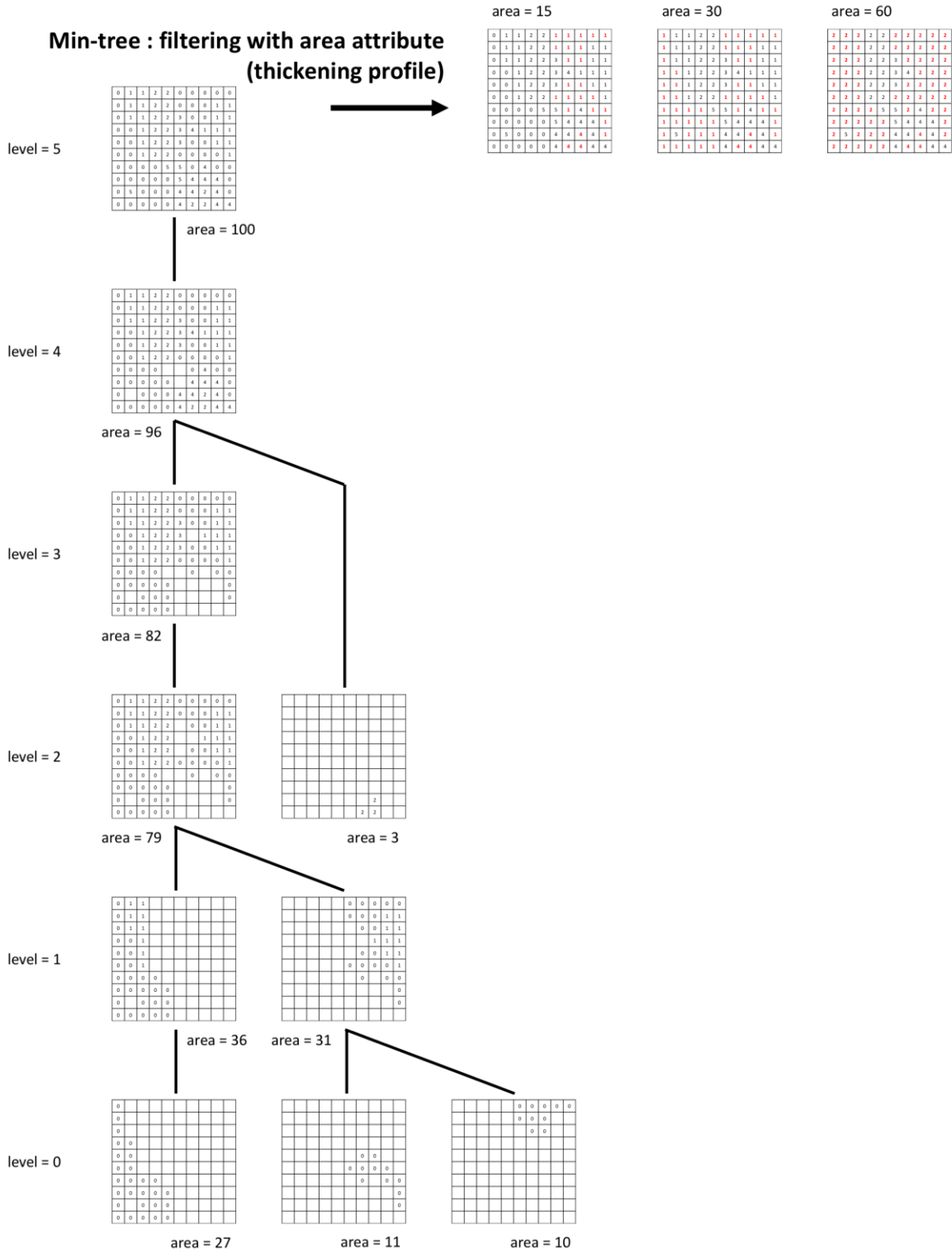

Figure 6: Thickening profile computed by filtering the min-tree with 3 thresholds {15, 30, 60} on the area attribute

Ad 10.

As it can be seen from the answers 7 and 9, the thinning and thickening profiles (built respectively by filtering the max and min-tree) do not highlight the same characteristics of the image. While both operate here on the area attribute, the former decreases grey levels of small (w.r.t. selected thresholds) bright components and the latter increases grey levels of small (w.r.t. selected thresholds) dark components. Filtered pixels are indicated in red in the two profiles: we can clearly see that the two profiles, even if they start with the same image (i.e., area threshold set as 0 or 1), contain different values and focus on different regions of the image.

Answers to Assignment 2b

Ad 1.

The max tree focuses on the bright regions and the min tree on the dark regions. Filtering each tree through an attribute filtering will remove the bright regions (for max tree) or dark regions (for min tree). (Images are stretched for the visualization purpose)
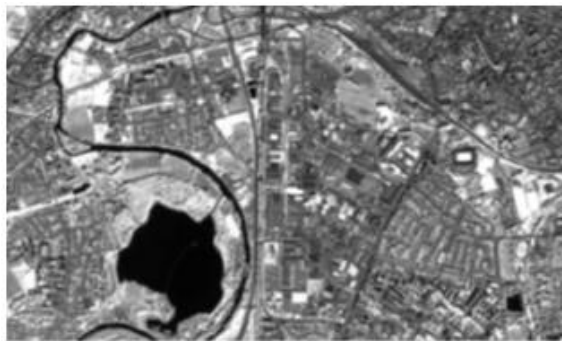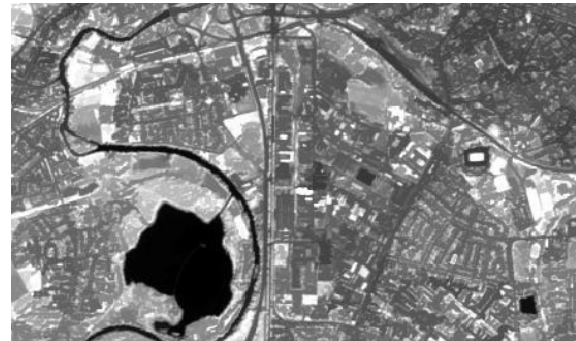
Original	Area threshold 100
Area threshold 500	Area threshold 1000
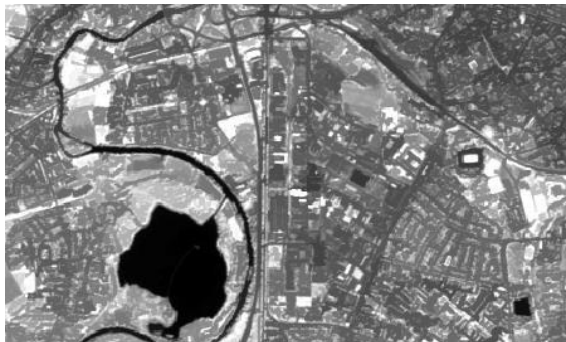
Figure 7: Min Tree.

| Area threshold 100 | Area threshold 500 |

| Area threshold 1000 | Area threshold 5000 |
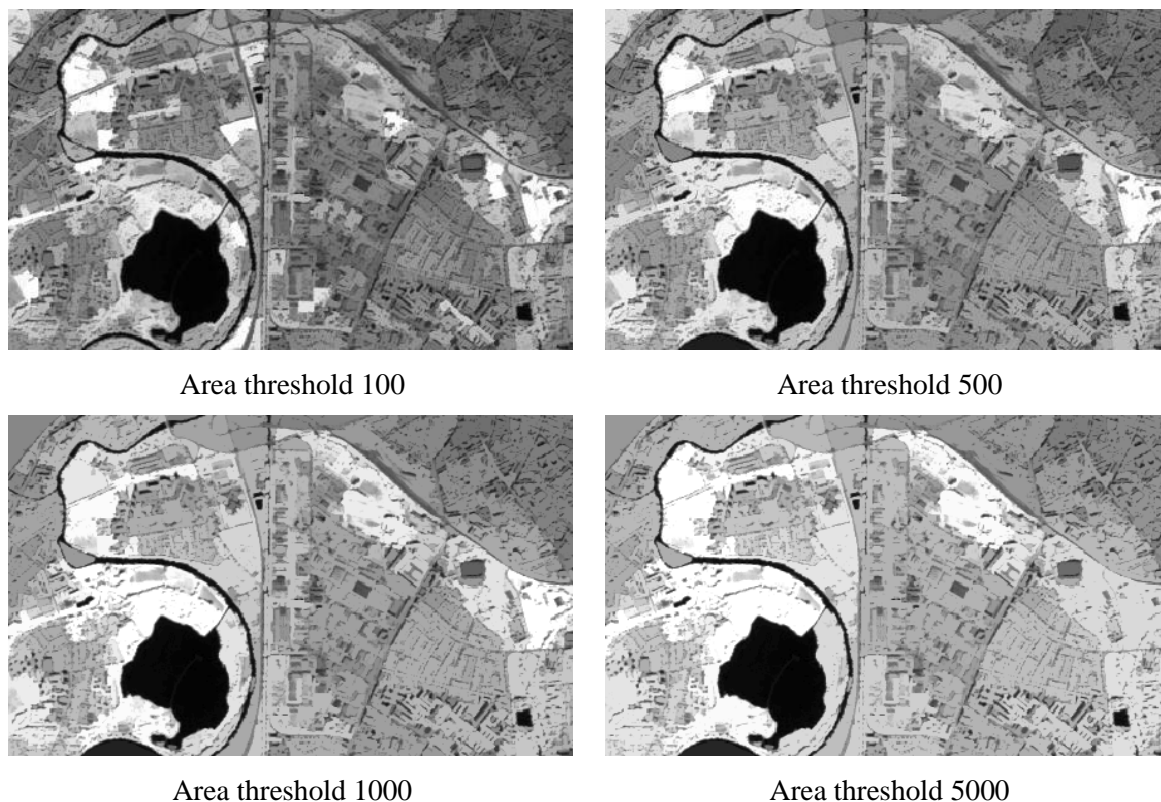
Figure 8: Max Tree.

Ad 2.

As the area thresholds are increasing from small value to larger value, small bright regions are merged into larger darker regions (lower grey value) in the max-tree. Conversely, in the min tree the small dark regions are merged into larger brighter regions (higher grey value).

Software tools
AP Tool
apGenerator.exe

Description
Command to generate attribute profiles in windows

Instructions

Instructions_for_APTool.pdf (Explanation how to generate the attribute profiles with area and standard deviation attributes)

Data
10m_cropped.tif (rectified satellite image)

areaTreshold.txt (area thresholds between 100 and 20000)

The software, data, description and instructions can be accessed at
https://github.com/JoaHoe/EduServ17Mat

Literature recommended to read

Introduction to Machine Learning (12 slides)
Machine Learning methods (45 slides)
#1: Randomized non-linear component analysis for dimensionality reduction of hyper-spectral images (4p.)
#2: Randomized PCA and MNF (31 slides, text to slides 7p.)
#3: Sparce HSIC Feature Selection (19 slides, article 14 p.,)

Supplementing literature

PerTurbo Classifier (2 articles, 4p. and 9p.)
Cross-validation (statistics) for test of model, comparison of methods.

Quiz

1) You are given a random sample from any distribution. What will be the value of entropy?

   Select one:
         a. Low value
         b. none of both
         c. High value.

2) For multi-class, SVM uses "one vs all" or "one vs one" approach. When the number of classes is 20, which approach is computationally less expensive in terms of computational time?

   Select one:
         a. One vs one
         b. Both of them
         c. One vs all

3) Random Fourier Features are sampled from which distribution?

   Select one:
         a. Beta distribution
         b. Gaussian distribution with unknown mean
         c. Gaussian distribution with mean zero
         d. Binomial distribution.

4) What is the name of the technique used in supervised classification to learn the function?

   Select one:
         a. Empirical risk minimization
         b. None of the above
         c. Bayes minimization
         d. min-max minimization

5) Random Fourier features are used to approximate by which of the kernel functions?

Select one:
        a. RBF kernel
        b. All of them
        c. Linear kernel
        d. Polynomial kernel.

6) Kernel function or matrix measures_____between the pair of data points.

Select one:
        a. Both of them
        b. Dissimilarities,
        c. Similarities
        d. None of both.

7) What kind of decision boundary will you use to distinguish between two classes?
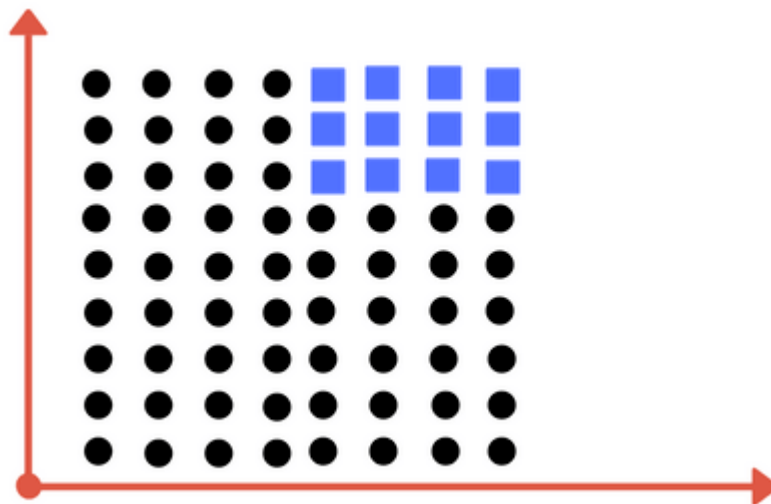
Select one:
        a. None of the above
        b. Linear
        c. Both
        d. Non-linear

8) _____ is used as the measure of disorder.

Select one:
        a. Entropy
        b. Euclidean distance
        c. Gini Index
        d. Both a and c

Assignment

1) Explain Hughes Phenomena or Curse of dimensionality.

2) How do you choose the attributes (features) to split in the decision tree. Explain also how the threshold function is chosen in the decision tree split?

3) What is known as 'overfitting' in supervised classification?

4) You are given very high dimensional data (i.e., the number of features is very large) and you are asked to design a classifier based on decision tree framework. Which of these classifiers will you use, Decision tree or Random Forest classifier, and why?

5) Explain what the support vector is in the SVM and what the kernel trick is in SVM.

6) Why is the SVM classifier with RBF kernel not scalable for large-scale data sets?

7) How do you sample the random Fourier feature coefficients and what kind of kernel can be approximated by the random Fourier features method?

8) How do you overcome "Curse of dimensionality" or "Hughes Phenomena"? Which methods will you use when you have and not have labelled training samples?

9) How is the out-of-bag estimate used to select the features in the Random Forest (RF) classifier?

10) What kind of class separability measure is considered in the Sparse HSIC feature selection method and how is this class separability measure different from the conventional class separability measure?

11) Random Fourier based PCA and MNF can exploit the non-linear information using all the pixels in the image, whereas conventional kernel PCA and MNF randomly samples few pixels from the image. Discuss some of the applications where random Fourier feature based PCA and MNF will be a better candidate than the conventional PCA and MNF method.

12) Perform classification using the RF classifier for the ISPRS dataset in Dropbox (either one of the data). Measure the accuracy of the performance. Now compute the attribute profiles using area attribute with min- and max-trees and perform RF classification. Discuss whether the inclusion of attribute profiles has increased the classification performance.

Solution to Quiz

Ad 1.   c. High value.
Ad 2.   c. One vs all
Ad 3.   c. Gaussian distribution with mean zero
Ad 4.   a. Empirical Risk Minimization
Ad 5.   a. RBF kernel
Ad 6.   c. Similarities
Ad 7.   d. Non-linear
Ad 8.   a. Entropy

Answers to Assignment

Ad 1.

Hughes Phenomena or curse of dimensionality arises when analysing the data in the high dimensional spaces with the supervised classification methods. With the fixed number of training samples, as the dimension of the data increases, the classification accuracy increases initially and beyond a certain point the classification accuracy starts to decrease. This is known as curse of dimensionality or Hughes Phenomena. This conveys that when the data lies in a high dimension, enough labelled samples are required to accurately estimate parameters of the underlying algorithm to produce accurate classification results.

Ad 2.

In Decision Tree, the information gain or Gini-index is computed for all the variables or attributes, and the attribute which has the minimum (maximum) information gain (Gini-index), is used to split the decision tree. Similarly, for the threshold, the information gain or Gini-index is computed over the possible range of thresholds and chosen is the one which has minimum information gain or maximum Gini-index. The selection of threshold and attributes are performed jointly.

Ad 3.

In supervised machine learning, 'overfitting' refers to a situation where the classifier exactly fits the training data (memorizes the training data) and produces high classification accuracy with training data but performs poorly on the unseen samples (testing samples) in the training stage. This is also known as poor generalization power of the classifier. This phenomenon mostly occurs when there are only limited (very small number of) training samples in the high dimensional data. A good classifier is considered to have good generalization power which performs well on training and testing data.

Ad 4.

In the high-dimensional data, generally the number of available labelled samples is small. So, the choice of the decision tree classifier will overfit the training data and have high variance in performance when the inputs are slightly changed. The Random Forest classifier is the best choice here, due to its ability to tackle the high dimensional data by randomly selecting the subset of attributes in each decision tree. It also reduces the variance by averaging the prediction from multiple Decision trees. Furthermore, the Random Forest classifier can be parallelized to handle the large-scale data.

Ad 5.

The closest points from the hyper-plane on both sides of the hyperplane are called support vectors and these support vectors determine the optimal hyperplane of the SVM. Kernel trick: When the data is not linearly separable, the data are mapped into higher dimensional space, where the data are linearly separable. The kernel method maps these data points implicitly to the higher dimensional space, without requiring knowing the mapping function. This is known as kernel trick.

Ad 6.

RBF kernel requires to compute pair-wise similarities between the data points, this involves cubic computational complexity. As the data size becomes larger, the computing of the kernel matrix

consumes too much time and memory. So, the SVM classifier with kernel function is not scalable for large scale datasets.

Ad 7.

Random Fourier feature coefficients are sampled from the Gaussian (normal) distribution with mean zero and variance $\sigma - 1$. The bias term is sampled from the uniform distribution between 0 to 2π. Random Fourier features can approximate the RBF kernel only.

Ad 8.

Dimensionality reduction methods either feature selection or feature extraction can be employed to overcome the curse of dimensionality or Hughes phenomena by reducing the dimension of the input data. When there is no labelled training samples, the dimensionality methods (unsupervised dimensionality reduction methods) such as PCA, MNF, RFFPCA, RFFMNF can be used, and when the labelled training samples are available supervised dimensionality reduction methods (feature selection) can be used, such as Random Forest, Sparse HSIC, etc.

Ad 9.

While training the individual Decision tree in the Random Forest, 1/3rd of the training samples is left out for the validation purpose, and these samples are called as out-of-bag samples. This out-of-bag samples can be used to rank the importance of the features in the feature selection with Random Forest. Ranking the importance features can be done as below. Calculate the out-of-bag accuracy of the original RF model. When the feature is used for split, randomly shuffle the samples and calculate the out-bag accuracy and compare (subtract) with original RF model out of-bag accuracy. Out-bag-accuracy decreases when the feature is more important.

Ad 10.

In the Sparse HSIC feature selection method, the class separability measure is designed based on comparing the similarities between the two distributions using kernel methods (that is using HSIC measure). The conventional class separability measure requires enough samples to compute the inverse of co-variance matrix, thus limited in the case of high dimensional datasets, and also the conventional method does not incorporate the non-linear relations in the data, as they do not have an implicit mapping function. The Sparse HSIC feature selection method is not sensitive to a small sample size, and effectively captures the non-linear relations in the data.

Ad 11.

Conventional kernel PCA and MNF method randomly samples few pixels in the image, so it might not be more informative, and neglect the objects which have small spatial extend. The applications where the RFF PCA and MNF might be a better candidate than kernel PCA and MNF are: - Classification problem - Target detection - Change detection.

Ad 12.

The classification accuracy of the ISPRS dataset (area #7) using Random Forest with original image and attribute profiles are shown below (training is performed using 500 samples per class). It is evident from the accuracy measures, that the inclusion of attribute profiles has increased the classification accuracy up to 6 % and shows the potential of attribute profiles. Further inclusion of attributes such as standard deviation, moment of inertia will improve the accuracy further.

Result of Random Forest Classification with original image:

```
$OA
        Accuray    LBound    UBound
[1,] 0.6801565 0.6797249 0.6805905

$conf
           Imp_surf Building Low_Veg    Tree     Car
Imp_surf    992032   132310   70998    3825  292729
Building    167972   962834   37791    4878  131117
Low_Veg      19679    23356  402313  179143   12239
Tree          5949     7756  409945  855713    2840
Car          25129    11512    3123     173   67203

$PA
             Accuray    LBound    UBound
Imp_surf   0.6649480 0.6641700 0.6657052
Building   0.7380346 0.7372811 0.7387886
Low_Veg    0.6318424 0.6306483 0.6330265
Tree       0.6673772 0.6665382 0.6681923
Car        0.6272447 0.6243337 0.6301369

$UA
             Accuray    LBound    UBound
Imp_surf   0.8193458 0.8186598 0.8200305
Building   0.8462481 0.8455844 0.8469101
Low_Veg    0.4353236 0.4343130 0.4363089
Tree       0.8198589 0.8191208 0.8205954
Car        0.1327787 0.1318457 0.1337156
```

Result of Random Forest Classification with attribute profiles (area attributes)

```
$OA
       Accuray    LBound    UBound
[1,] 0.74342 0.7430586 0.7438249

$conf
           Imp_surf Building Low_Veg    Tree     Car
Imp_surf   1076895   142477   62626    4836  205060
Building    137701  1033622   28934    3388  100947
Low_Veg      14387    22012  420978  171220    8133
```

```
Tree          6366     7982  294395  970959    2501
Car          16175     6560    1515     157   82733

$PA
             Accuray    LBound    UBound
Imp_surf   0.7218308 0.7211249 0.7225494
Building   0.7922952 0.7915985 0.7929907
Low_Veg    0.6611562 0.6599647 0.6623181
Tree       0.7572584 0.7565158 0.7580000
Car        0.7721953 0.7696769 0.7747001

$UA
             Accuray    LBound    UBound
Imp_surf   0.8604669 0.8598590 0.8610732
Building   0.8523642 0.8517320 0.8529948
Low_Veg    0.5207237 0.5196210 0.5218126
Tree       0.8439012 0.8432372 0.8445637
Car        0.2071567 0.2059017 0.2084156

$F1
                  F1
Imp_surf   0.7850754 0.7845909 0.7855612
Building   0.8212327 0.8207591 0.8217057
Low_Veg    0.5825967 0.5818065 0.5834006
Tree       0.7982356 0.7977309 0.7987396
Car        0.3266761 0.3253854 0.3279486
```

Software tools

R-programs:

bh_classifiedimageread.R

bh_confusionmat.R

bh_confwriteexcel.R

bh_datapartitiontolist.R

bh_decisiontreeclassification.R

bh_EnviHeaderWrite.R

bh_ENVIWrite.R

bh_ISPRS_labelcoding.R

bh_ksvmcrossvalidation.R

bh_labelcoding.R

bh_loadLib.R

bh_loadparallel_Lib.R

bh_normalize_par.R

bh_normalize.R

bh_perturboclassification.R

bh_perturbocrossvalidation.R

bh_randomforest.R

bh_randomsubsetsamples.R

bh_rasterclassificationMap.R

bh_rbf.R

bh_readsamples.R

bh_rff.R

bh_statistics.R

bh_svmclassification.R

ISPRS_classifiction.R

MainFile_SVM_PreTrainedImgClassification.R

MainFile_SVMCluster_AP.R

MainFile_System.R

MainFille_SVMCluster.R

Instructions

R Programs for Classification.docx

ReadME.txt

Description of programs

MainFile_System.R - the main file to run for the classification experiments, it includes all the classifiers, hyper-parameter tuning and accuracy assessment methods.

bh_loadLib.R – library to be installed.

bh_decisiontreeclassification.R - Decision Tree classifier

bh_randomforest.R - Random Forest classifier

bh_svmclassification.R - SVM classifier

bh_perturboclassification.R - PerTurbo classifier

bh_rff.R - Large scale SVM using random Fourier Features

ISPRS_classifiction.R - Using the ISPRS dataset. The RGDAL package is used to read the tif images. The output images can be saved either using raster package in "tif" format or in ".hdr" envi format.

bh_confusionmat.R - Calculation of classification accuracy (overall, producer and user accuracy incl. confidence interval).

Data

ISPRS data set

top_mosaic_09cm_area1_Img.tif - ortho-image of area #1

top_mosaic_09cm_area1.tif - reference map of area #1

ortho_top_mosaic_09cm_area7.tif - ortho-image of area #7

GTS_top_mosaic_09cm_area7.tif - reference map of area #7

RCD30 data

train_all_6.csv - training sample for attribute generation

Source code of the programs, data and instructions are released at
https://github.com/JoaHoe/EduServ17Mat
Licensed under a Creative Commons Attribution Share Alike 4.0 International License

Literature recommended to read:

#1: Generation and assessment of urban land cover maps using high-resolution multispectral aerial cameras, 11 p.

#2: Generating topographic map data from classification results, 24 p.

#3: Classification of land cover using decision tree. 9 p.

Supplementing literature

Assessment of the thematic accuracy of land cover maps, 8 p.

Hough transform (9 sections)

Hyper-parameter tuning and cross validation (6 slides)

Cross-validation (statistics) (12 sections)

Quiz

1) The thematic accuracy of a single class is assumed to be 80%. The 95% confidence interval should have a half-width of d=10%, i.e., the accuracy is between 70% and 90%. Is the required sample size 61 or 122 or 244 points?

2) How many unknowns must be determined for a building with 8 corners to achieve parallel and orthogonal line segments which form the outline of the building? 8 or 16 or 9 unknowns?

3) The geometric accuracy of extracted and enhanced buildings is derived from 73 points and has a standard deviation of 1.0 m. The upper bound of the 95% confidence interval of the calculated standard deviation is 1.2 m or 1.3 m or 1.4 m ?

4) A straight line in the 2D Euclidean space is represented in the Hough space by a sinusoidal curve or a sinusoidal surface or a point?

5) Cross validation (CV) is used for the assessment of thematic accuracy or geometric accuracy or tuning of parameters/models?

Assignment

1) Calculate the overall accuracy, the user's, and producer's accuracy for class b ('building') using the attached error matrix.

reference class	b	h	g	r	t	w	row Σ
b	90	0	1	0	0	0	91
h	0	71	17	1	1	1	91
g	3	8	74	5	0	1	91
r	5	2	0	82	1	1	91
t	10	4	6	0	71	0	91
w	8	8	8	43	0	24	91
col Σ	116	93	106	131	73	27	546

Error matrix of a land cover map derived by DT (b='building', h='hedge & bush', g='grass', r='road & parking lot', t='tree', w='wall & car port')

2) Explain the difference to the results for the accuracy measures in (Höhle, 2015).

3) Explain the cross-validation method.

4) Determine the sample size for a map with 6 classes. Assume accuracy of class=5%, level of confidence=95%. (In the given equation, the class proportion is assumed to be 50%).

5) After the Hough transform the parameter combination ($\theta=60°$, $\rho=80$ pixel) got a high number of votes. What is the slope and the intercept (intersection with the y-axis) of this line segment of a building outline?

6) Why should the line not be determined by linear regression?

7) The point cluster representing a line consists of three points (pixels): $(x_i, y_i) = (0,0)$, $(1,3)$, and $(2,12)$. Calculate the line parameters (θ, ρ) by least-squares adjustment.

8) Describe how the generation of orthogonal and parallel lines at buildings may be accomplished in automated mapping.

9) Intersect line 1 ($\theta_1=150°$, $\rho_1=0.5$ pixel) with line 2 ($\theta_2=150°$, $\rho_2=5$ pixel).

10) Describe how the point clusters representing line segments of a building outline are separated and how the sequence of the line segments is determined. (View/run the attached R-programs depicted in the attached flowchart "Extracting point clusters").

Solutions to Quiz

Ad 1.

The sample size may be calculated by the program "samplesize_2.r". Explanations to the theoretical background of the program are given in the article "Höhle & Höhle, 2013".
The right answer is 61 points.
(cf. R-function: binomSamSize::ciss.binom(p0=0.8, d=0.1, alpha=0.05, method="lrt")

Ad 2.

The correct answer is 9 unknowns (8 for line parameters 'ro', and 1 for the line parameter 'theta')

Ad 3.

Equation 27 in: Höhle, J. "Automated mapping in urban areas", 2018

$$P\left\{ \sqrt{\frac{ms^2}{\chi^2_{1-\alpha/2,m}}} < \sigma < \sqrt{\frac{ms^2}{\chi^2_{\alpha/2,m}}} \right\} = 1 - \alpha$$

m=n-1

n=73, *m= 72,* α=0.05, α/2=0.025, s=1.0 m

$\chi^2_{0.025,72} = 50.43$ (determined by R-function qchisq(p, m, lower.tail=TRUE), p=0.025, m=72)

The right answer is 1.2 m (cf. R-program "ci_sigma.r")

Ad 4.

The correct answer is 'point'. In the reference 'Wikipedia, 2017. Hough transform' (Example 2) two lines are depicted as points in the Hough space.

Ad 5.

The correct answer is: "For tuning parameters/models".
The CV method is used for tuning of parameters/models, e.g., the hyper-parameters C and γ in the support vector machine (SVM) method of Machine Learning. More details can be found in the reference 'Hyper-parameter tuning and cross validation'. In principle, the assessment of the thematic accuracy of classification results must be carried out by independent data, i.e., they should not be used in the training of the classifier.

Answers to Assignment

Ad 1.

$$overall\ accuracy = \frac{412}{546} = 0.755$$

$$user´s\ accuracy_b = \frac{90}{91} = 0.989$$

$$producer´s\ accuracy_b = \frac{90}{116} = 0.776$$

Ad 2.

The calculated accuracy measures in (Höhle 2015) used weights (w_i) for the counts

$$\bar{n}_{ij} = w_i \cdot n_{ij}$$

$$w_i = \frac{N_i}{N} \cdot \frac{n}{n_{i+}}$$

Ad 3.

In the cross-validation method, the set of reference data is split into cells for training of a DT and cells for validation. There exist several variants, e.g., "Leave one out cross-validation". At this variant, the class label of one cell is predicted by a classifier, which is derived from the (n-1) cells. The reference value is not used for the classification of this cell. The procedure is repeated n times resulting in n different classifiers. From the assigned class values an error matrix can be formed and approximate accuracy measures may be derived. Cross validation is usually used to test the applied model (selection of variables or parameters) or to compare different methods of classification.

Ad 4.

With k = 6, α = 0.05 → B = 6.9605 (calculated by R-function 'qchisq')
qchisq(df=1, 1-0.05/6)

At a desired class precision (b = 0.05):
n = 696 and n_c = 696/6 = 116.

Ad 5.

Y = -ctgθ·x+ρ/sinθ
slope = tanα = -ctgθ = -1/tanθ = -1/tan60°=-1/√3=-0.5774→α = -30°
intercept = ρ/sinθ = 80/sin60° = 80·2/√3=92.4 pel

alternative solution for α:
α = θ-90=-30 or
α = θ+90+180= 330.

Ad 6.

In linear regression the residuals in the y-coordinate are minimized. This does not work for vertical lines. The residuals should be perpendicular to the line. Therefore, the Hesse normal form of a line is a better solution.

Ad 7.

$(x_1, y_1) = (0, 0)$, $(x_2, y_2) = (1, 3)$, and $(x_3, y_3) = (2, 12)$
xs=∑x_i/n=1
ys=∑y_i/n=5
x'=x_i-xs
y'=y_i-ys

i	x_i	x'	x'^2	y_i	y'	y'^2	x' y'
1	0	-1	1	0	-5	25	5
2	1	0	0	3	-2	4	0
3	2	1	1	12	+7	49	7
Σ	**3**	**0**	**2**	**15**	**0**	**78**	**12**
	xs=1			ys=5			

$a = 2$

$b = 2 \cdot 12 = 24$

$c = 78$

$\tan(2\theta) = b \cdot (a-c) = 24/(2-78) = -0.31579$
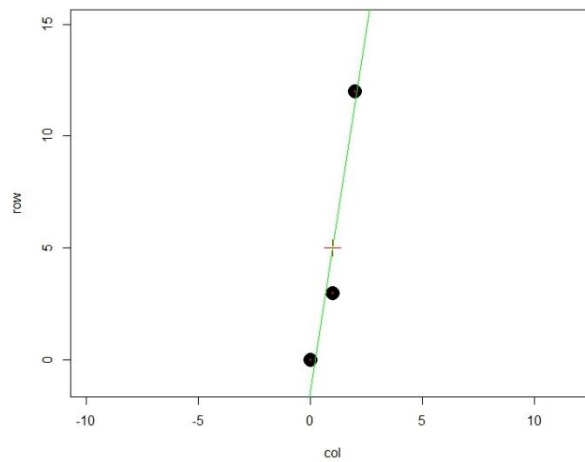
$2\theta = -17.5256°$

$\theta = -8.76.28° = +171.2372°$

$\rho = xs \cdot \cos\theta + ys \cdot \sin\theta$

$\rho = 1 \cdot 0.988328 + 5 \cdot (-0.152344) = +0.22661$

$\rho = 0.23$

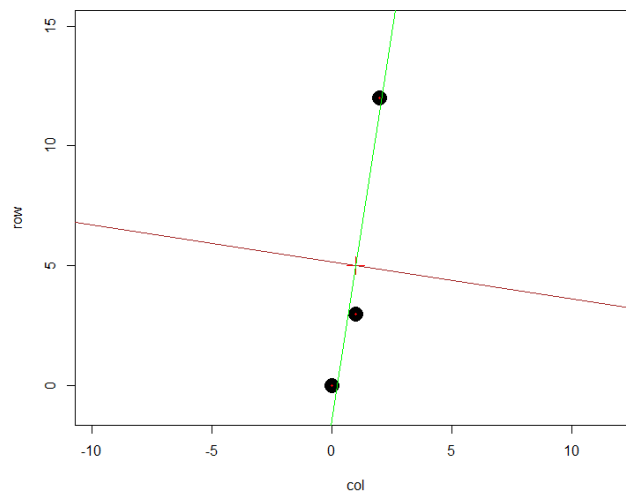or alternative solution (θ_2, ρ_2): brown line

$\theta_2 = \theta + 90° = -8.76.28° + 90° = 81.2372°$

$\rho_2 = 1 \cdot 0.152344 + 5 \cdot 0.988328$

$\rho_2 = 5.09$

The correct solution is the green line.

Ad 8.

The generation of building outlines with orthogonal and parallel lines is achieved by:
- intersecting successive lines
- the weighted average for the angle (θ_{av})
- determination of the ρ-values by least-squares adjustment

Ad 9.

The coordinates of the intersected point may be calculated by equations 9-13 in (Höhle 2017). They can be modified to ease the calculation.

$$\begin{pmatrix} x_{P_k} \\ y_{P_k} \end{pmatrix} = \begin{pmatrix} q_1 & q_2 \\ q_3 & q_4 \end{pmatrix} \begin{pmatrix} \rho_i \\ \rho_{i+1} \end{pmatrix}$$

$$k_1 = \sin\theta_i - \cos\theta_i \cdot \tan\theta_{i+1}$$
$$k_2 = \cos\theta_{i+1} \cdot \tan\theta_i - \sin\theta_{i+1}$$

$$q_1 = \frac{1}{\cos\theta_i} - \frac{\tan\theta_i}{k1}$$

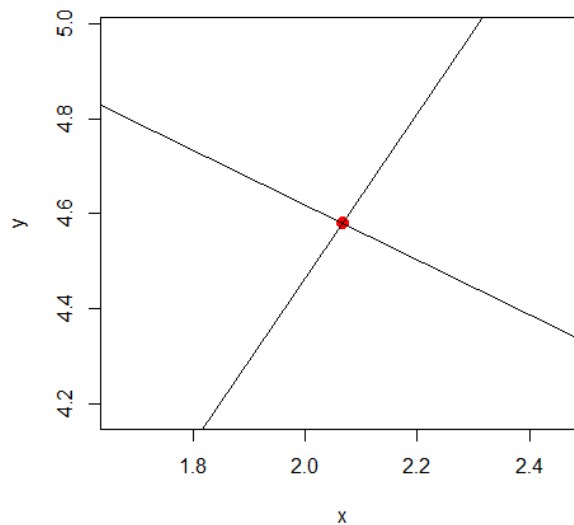$$q_2 = \frac{\tan\theta_i}{k2}$$

$$q_3 = \frac{1}{k1}$$

$$q_4 = \frac{-1}{k2}$$

With ($\theta_1 = 150$, $\rho_1 = 0.5$) and ($\theta_2 = 60$, $\rho_2 = 5$) the coordinates are:

$$\begin{pmatrix} x_P \\ y_P \end{pmatrix} = \begin{pmatrix} -0.866 & 0.5 \\ 0.5 & 0.866 \end{pmatrix} \begin{pmatrix} 0.5 \\ 5 \end{pmatrix}$$

$x_P = 2.1\,\text{pixel}$, $y_P = 4.6$ pixel

Ad 10.

The separation of the point clusters representing line segments of the building outline is done by means of the Hough transform. All pixels of the extracted 'connected components' image of one building are transformed into the parameter space. The parameters of a line are the angle (theta) and the distance from the origin (ro). All pixels situated close to a line segment are collected in a cell of the Hough matrix representing the parameter space. The cell with the highest score is the reference line (segment). The big matrix must be further analysed to find out which lines belong to the outline of the building. The program ('line detection.R') accomplishes this task. The Hough matrix is displayed. The brightest spot is the reference line. Several point clusters representing a line are extracted and then analysed. The point clusters representing a line segment of the outline are found by the condition of parallelism, rectangularity, and the length of a line. Another problem to be solved is the sequence of the line segments representing the outline of the building. It is solved by calculating the centre of the detected line segment. The angle from the centre of the building to the centre of the line segment is calculated and arranged in a decreasing order. An alternative approach traces the outline of connected pixels and searches for plotted points representing the detected line segments. Successive and adjusted lines are then intersected. The sequence of line segments is checked by a test. The R-program 'sequence of lines.R' displays the sequence of line segments.

Software tools

"EBImage"
Open-source software package, R-package
Details are contained in: EBImage.pdf (bioconductor.org)

'spatstat'
Open-source software package, R-package
Details are contained in: spatstatQuickref.pdf (spatstat.org)

'samplesize_1.r' and 'ci_sigma.r'
R-programs to calculate sample size and confidence interval. The first program realizes a worst-case scenario by means of a simple formula published in (Congalton and Green, 2009); the second program calculates the confidence intervals for the standard deviation. Details can be found in (Mikhail, 1976).

'samplesize_2.r'
R-package. A suite of functions to compute confidence intervals and necessary sample sizes for the parameter p of the Bernoulli B(p) distribution under simple random sampling or under pooled sampling.

'orthogonal_buildings.r'; 'Hough transform for line detection.r'; 'outline_of _building.r'
R-programs. A selection of programs for solving the assignment of Module 4.

Instructions
Installation of R + RStudio.pdf
Flowchart of R-programs for extracting point clusters.pdf

Data
LCM_cart_enh_buildings.jpg

Source code of the programs, instructions and data are released at
https://github.com/JoaHoe/EduServ17Mat
Licensed under a Creative Commons Attribution Share Alike 4.0 International License

'buildenh' (2023)
Open-source software package in R language.
Details are contained in the article: 'buildenh' - a collection of R-scripts for the cartographic enhancement of buildings generated by classification of remote sensing imagery.

Source code is released at https://github.com/JoaHoe/buildenh_v1.3
Licensed under a Creative Commons Attribution Share Alike 4.0 International License

Module 5

Literature recommended to read:
#1: Automated mapping of urban areas, 20 p.
#2: Attribute profiles on derived features for urban land cover classification, 11 p.
#3: Related work in automated mapping of topographic data, 5 p.
#4: Assessment of the thematic accuracy of land cover maps, 8 p.
Enhancement to raster data (8 slides)
Detection of small woody features (9 slides)

Supplementing literature:
Generation of Land Cover Maps Using High-Resolution Multispectral Aerial Cameras, 6p.

Quiz

1) The reliability of the thematic accuracy is determined by the confidence interval (CI). How big is the 95% CI for buildings in the example with the RCD30 data in Reference #1?

   5% or 2.5% or 1%

2) Which classifier and attribute combination achieved the best overall accuracy for the example with the ISPRS data in Reference #2?
   79.02% or 68.94% or 77.63%

3) What was the improvement in overall accuracy when using combination of simple + multi-scale attributes instead of simple attributes with ISPRS data and DT classifier only?
   6.85% or 7.12% or 0.87%

4) Which combination of simple attributes achieved the best overall accuracy in the investigations of Reference #2?
   Orthoimage + vegetation index (NDVI) or orthoimage + height above ground (nDSM) or orthoimage + NDVI + nDSM?

5) The achieved geometric accuracy (RMSEx,y) in the example of article #1 was:
   3 or 11 or 16 GSD?

Assignment 1

1) When should the results of classification further be enhanced?
2) How is the geometric accuracy of vector data assessed?
3) Describe the steps in the enhancement of a raster map.
4) Describe the method you would use for automated 2D mapping of buildings in scales bigger than 1:25000. Discuss data source, attributes, classification method, cartographic enhancement (raster & vector), required accuracy, and economy. Reference: Article "Automated mapping of buildings".
5) How can the quality of the adjustment of the building corners be evaluated? (View/run 'adjustment_of_corner_coordinates.r')

Assignment 2

This assignment is voluntary; however, it is recommended to learn the practical application of attribute profiles. Programs and data for this assignment are accessed by a Dropbox link in "Moodle".

Question 1:
a) Compute the NDVI of the input image.
b) Compute the attribute profiles of the NDVI image using area attributes with max tree and area thresholds 1000, 2500, 5000, and 10000.
c) Load the original image, NDVIAP, and the Ground Truth image
d) Perform RF classification.

Question 2:
a) Load the original image, Ground Truth image, and NDVIAP from (b) of question 1
b) Use the NDVIAP from the question 1 to compute differential attribute profiles (DAP)
c) Concatenate OrgImg, NDVI, NDVIDAP, and perform RF classification
d) Display the detected woody feature map
e) Discuss the performance difference between question 1 and question 2

Solutions to Quiz

Ad 1.

The confidence interval (CI) is given by an upper and a lower bound (here: 95% and 100%). The difference between UB and LB is 5% for buildings in this example (RCD30 data). The 95% CI means that at repetition of new assessments, the true value of the thematic accuracy will in 95% of all assessments lay in this interval.

Ad 2.

The best overall accuracy is 79.02%. It was achieved with SVM, OrthoImg+DSM_AP using the ISPRS data.

Ad 3.

The correct answer is: 6.85%

Ad 4.

The correct answer is: Orthoimage + NDVI + nDSM.

The different classifiers yield nearly the same result (which is an important result in the investigation). When using the average of the overall accuracies of all classifiers the values are:

96.81% (orthoimage + NDVI + nDSM),
96.39 % (orthoimage+nDSM) and
82.70% (orthoimage+NDVI).

This means, that the use of nDSM instead of the NDVI achieves a considerable improvement (13.69%) for the overall accuracy. The gain when using all three attributes is 0.42% only which also is an important result.

Ad 5.

Correct solution: 11 GSD
GSD= 9cm
RMSExy = 1m = 1/0.09 GSD = 11.1 GSD

Remark: The data material with GSD=9cm has a high resolution. In the case of automated mapping: The GSD should be larger to improve the economy.

Answers to Assignment 1

Ad 1.

Enhancement of the classification result becomes necessary when topographic map data are the goal. Also, when specified values for the thematic accuracy are not obtained.

The classification results may contain inhomogeneous areas and small objects. Therefore, a generalization of the content is necessary. Small objects must be removed, and area objects must have a homogeneous colour. Such an enhancement is still a raster map. To generate vector data for topographic databases, the man-made objects must be represented by straight lines and regular curves. Buildings must have parallel and orthogonal lines. A higher geometric accuracy can also be achieved when using least-squares adjustment procedures.

Ad 2.

A first evaluation can be made by plotting the outlines on top of the DSM-based orthoimage. Blunders will be visible.

To access the geometric accuracy of vector data, accurate checkpoints must be available. Suitable checkpoints must be well-defined in the reference and in the enhanced classification. Suitable points are, e.g., the corners of buildings (if the classification is carried out by means of a DSM-based orthoimage). Reference coordinates may be determined by means of the reference map or the DSM-based orthoimage. The standard accuracy measures are the root mean square error (RMSE), the mean (μ), and the standard deviation (σ). They are determined for both coordinates (x, y).

The number of checkpoints should be n > 50 and the accuracy of the checkpoints should be 0.1σ of the required accuracy for the derived points. Gross errors must be eliminated. They are defined as $\Delta > 3\sigma$, their number (NB) should be quoted. The planimetric accuracy may also be quoted as a distance between the reference and the corner of the enhanced building.

Ad 3.

The classification has separated the image content into classes representing topographic objects. The extracted class "building", e.g., is processed to an enhanced raster image in a few steps. Firstly, some smoothing of the building's shape is achieved using morphological operations like dilation

and erosion. The outlines of the buildings can then be obtained by adaptive thresholding. A moving window filters the image. The width and height of the window and the offset from the averaged value must be specified, e.g., 5x5 pixels and 0.01, respectively. The buildings are then completely filled by pixels of the same colour. The connected sets of pixels are labelled. Geometric features like the area of the individual objects are derived. Objects smaller than a threshold value of the area are removed. The generated raster image of the class 'building' can be combined with other classes. It may also be used as input to create a vector map.

Ad 4.

Data source: Metric aerial images of four bands (R, G, B, NIR), GSD < 10 cm
Attributes: DSM generated by dense matching, derived nDSM and σ_{DSM}, NDVI,
Classification method: Decision Tree
Cartographic enhancement (raster): Morphological operations, labelling of objects,
Cartographic enhancement (vector): Extraction of straight lines, determination of line sequence, intersection of corner points, generation of orthogonal and parallel lines.
Thematic accuracy: At overlaps of derived buildings to reference > 80%: correctness=100%, completeness/quality=78%
Geometric accuracy: Achieved: $RMSE_x$=1.4m, $RMSE_y$=1.1 m, required: $RMSE_x = RMSE_y < 1.0$ m
Economy: Inexpensive tools (camera, computer), multi-use data (orthoimage, DEMs), carrier: airplane (flying height up to 13 km).

Remark: The given answers are based on the method described in (Höhle, 2019). Improvements are required with respect to accuracy and speed of operation. With the used GSD, objects of about 30 x 30 cm^2 are visible in orthoimage and can be detected and plotted by automated methods. When using bigger GSDs, the economy of the operation can be improved but the detection of small objects will suffer.

Ad 5.

The calculation of the building corners is done in two steps: The orientation angle is determined by the weighted average; the second step is a least squares adjustment where the residuals in the corner coordinates are minimized. The quality of the corner coordinates is evaluated by the standard deviation of the residuals (σ_r) and the maximal error (r_{max}). In the example (building #13, RCD30 data, orthoimage #57), these values are $\sigma_r = 1.1$ pixel and $r_{max} = 1.5$ pixel using the program 'adj_of_corner_coordinates.r'.

Answers to Assignment 2

To compute NDVI DAP, you need to insert the following lines in the SIRS_SingleFile_RF_Train_Test.R, after loading NDVIAP image:

```
for (i in 2:sz[2])
{
AreaAPImg[,i]<-AreaAPImg[,1]-AreaAPImg[,i]
}
```
AreaAPImg[,1] - indicates the NDVI image
AreaAPImg[,2] - indicates the NDVI AP with threshold 1
AreaAPImg[,3] - indicates the NDVI AP with threshold 2
and so on.

The performance between the NDVI AP and NDVIDAP is similar, however NDVI DAP can reduce number of false positives in the classification map.

Software tools

R-program for the derivation of final corner coordinates
adjustment_of_corner_coordinates.r

R-program for the enhancement of buildings in raster format
cart_enhance_raster.r

Source code of both programs is released at https://github.com/JoaHoe/EduServ17Mat
Licensed under a Creative Commons Attribution Share Alike 4.0 International License

R-program for the detection of small woody features
SIRS_SingleFile_RF_Train_Test.R' (detection of hedgerows and trees).

Source code of the program is released at https://github.com/JoaHoe/EduServ17Mat
Licensed under a Creative Commons Attribution Share Alike 4.0 International License

Instructions

Details for use of the program are in 'description_woody_detection.docx' and in (Merciol et al., 2019).

Data

Data for the enhancement of buildings

b13_coord_appr.txt (approximate coordinates of corner points, building #13)
theta_av.txt (orientation angle of main line, building #13)
b13_center.txt (xy-coordinates of PC centre, building #13)

The data can be accessed at https://github.com/JoaHoe/EduServ17Mat

SWF_570d_ext_v1.tif (satellite image)
SWF_SAMPLE_ext_v1.tif (ground truth)
SWF_570d_ext_v1.tfw (TIFF world file)

The data are described in 'description_woody_detection.docx' and (Merciol et al., 2019).

The data can be accessed at https://github.com/JoaHoe/EduServ17Mat