

Friedrich Schiller University Jena  
Faculty for Chemistry and Earth Sciences  
Department of Geography  
GIScience group

**Automatic landslide detection using  
high-resolution LiDAR DTM data.  
An object-oriented approach with  
open-source software in Burgenland, Austria.**

THESIS  
To obtain the degree of  
**Master of Science (M.Sc.)**

Submitted by  
**Raphael Knevels**  
born on 20 February 1990  
in Hamm, North Rhine-Westphalia, Germany

Advised by  
**Dr. Helene Petschko**

Jena, April, 2017

## ABSTRACT

With high-resolution airborne Light Detection and Ranging (LiDAR) data more commonly available, many studies have been performed to facilitate the detailed information on the earth surface and to analyze its limitation. Specifically in the field of natural hazards, digital terrain models (DTM) have been used to map hazardous processes such as landslides mainly by visual interpretation of LiDAR DTM derivatives. However, new approaches are striving towards automatic detection of landslides to speed up the process of generating landslide inventories. These studies usually use a combination of optical imagery and terrain data, and are designed in commercial software packages such as ESRI ArcGIS, Definiens eCognition, or MathWorks MATLAB.

The objective of this thesis was to investigate the potential of open-source software for automatic landslide detection based only on high-resolution LiDAR DTM derivatives in a study area within the federal state of Burgenland, Austria. The study area is very prone to landslides which have been mapped with different methodologies in recent years. The free development environment R was used to integrate open-source geographic information system (GIS) software, such as SAGA (System for Automated Geoscientific Analyses), GRASS (Geographic Resources Analysis Support System), or TauDEM (Terrain Analysis Using Digital Elevation Models).

The implemented geographic-object-based image analysis (GEOBIA) consisted of (1) derivation of land surface parameters, such as slope, surface roughness, curvature, or flow direction, (2) finding optimal scale parameters by the use of an objective function, (3) three-level multi-scale segmentation, (4) classification of landslide parts (scarp, flanks, body) by k-means thresholding, (5) assessment of the classification performance using a pre-existing landslide inventory, and (6) post-processing analysis for the further use in landslide inventories. The results of the developed open-source approach led to landslide detection accuracies of approximately 68.89% in correctly identified landslide areas, and 65.54% in the correctly recognized total number of inventory landslides. Moreover, the additional detection of 22.31 ha of landslide area outside the inventory, shows the great potential of the algorithm to support the manual mapping method.

## Table of Content

<b>Table of Content .....</b>	i
<b>List of Figures.....</b>	iv
<b>List of Tables .....</b>	vi
<b>List of Acronyms .....</b>	vii
<b>1 Introduction.....</b>	1
<b>2 Theoretical Background and Research Question.....</b>	3
2.1 Fundamentals.....	3
2.1.1 LiDAR .....	3
2.1.1.1 Definition .....	3
2.1.1.2 Mode of Operation.....	3
2.1.1.3 Advantages, Limitations and Applications.....	5
2.1.2 Geomorphometry and Land Surface Parameter.....	6
2.1.2.1 Definition .....	7
2.1.2.2 DTM .....	8
2.1.2.3 Slope and Curvature .....	9
2.1.2.4 D-Infinity Flow Direction and Accumulation.....	10
2.1.2.5 Sky-View Factor .....	11
2.1.2.6 Openness.....	12
2.1.2.7 Surface Roughness.....	13
2.1.3 Shape Metrics.....	14
2.1.4 Textures.....	16
2.1.5 Landslides.....	18
2.1.5.1 Definition .....	19
2.1.5.2 Types .....	19
2.1.5.3 Causes and Triggers .....	21
2.2 State of the Art.....	22
2.3 Research Question .....	25

<b>3 Study Area and Related Data .....</b>	<b>27</b>
3.1 Study Area in Burgenland .....	27
3.2 Digital Terrain Model.....	29
3.3 Landslide Inventory.....	29
<b>4 Methodology .....</b>	<b>32</b>
4.1 Theory of Automatic Landslide Detection.....	32
4.1.1 Object-Oriented Image Analysis.....	32
4.1.1.1 Introduction .....	32
4.1.1.2 Segmentation Algorithms .....	33
4.1.1.3 Derivation of Seeds .....	34
4.1.1.4 Optimization of Scale Parameters .....	35
4.1.2 Detection of Landslides: Idea and Workflow.....	39
4.1.2.1 Segmentation Level 1: Detection of Scarp.....	41
4.1.2.2 Segmentation Level 2: Detection of Flank .....	43
4.1.2.3 Segmentation Level 3: Detection of Landslide Body and Cleaning	
.....	44
4.1.3 K-Means Thresholding .....	46
4.1.4 Accuracy Assessment.....	47
4.2 GIS Open Source Software .....	48
4.2.1 R.....	48
4.2.2 GRASS GIS .....	48
4.2.3 SAGA GIS.....	49
4.2.4 Additional Software .....	49
4.3 Implementation of Functions in R.....	50
4.3.1 Segmentation Function .....	51
4.3.2 Objective Function .....	52
4.3.3 Bounding Box in Flow Direction .....	53
4.3.4 Object Orientation.....	54
4.3.5 Neighbor Operations.....	54
4.3.6 Texture in Flow Direction .....	57
4.3.7 Other Functions .....	57

<b>5 Results.....</b>	<b>59</b>
5.1 Landslide Detection .....	59
5.1.1 Segmentation Level 1 .....	59
5.1.2 Segmentation Level 2 .....	63
5.1.3 Segmentation Level 3 .....	68
5.2 Accuracy Assessment and Comparison.....	76
<b>6 Discussion .....</b>	<b>80</b>
6.1 Landslide Detection .....	80
6.2 Input Parameters .....	84
6.3 Usability of Open Source Products and <i>Lslide</i> Package.....	87
<b>7 Conclusion and Perspective .....</b>	<b>90</b>
<b>8 References .....</b>	<b>91</b>
<b>Appendix.....</b>	<b>102</b>
A. Data of Segmentation Levels .....	102
B. Results of Objective Function .....	104

## List of Figures

Figure 2-1: Concept of Airborne LiDAR (HÖFLE & RUTZINGER 2011:4).....	4
Figure 2-2: Illustration of LiDAR returns at a tree (CAMPBELL & WYNNE 2011:251). .....	5
Figure 2-3: LiDAR point cloud from Central Park in New York City (O'NEIL-DUNNE). .....	6
Figure 2-4: Land surface parameter: digital terrain model (OWN REPRESENTATION).....	8
Figure 2-5: Land surface parameter: normalized height (OWN REPRESENTATION).....	9
Figure 2-6: Land surface parameter: slope (OWN REPRESENTATION).....	10
Figure 2-7: Land surface parameter: maximum curvature (OWN REPRESENTATION).....	10
Figure 2-8: Land surface parameter: D-Infinity flow direction and accumulation (OWN REPRESENTATION).....	11
Figure 2-9: Land surface parameter: maximum curvature (OWN REPRESENTATION).....	12
Figure 2-10: Land surface parameter: openness (OWN REPRESENTATION).....	13
Figure 2-11: Land surface parameter: surface roughness (OWN REPRESENTATION).....	14
Figure 2-12: Objects of different shape (OWN REPRESENTATION).....	16
Figure 2-13: Texture entropy in flow direction (OWN REPRESENTATION).....	18
Figure 2-14: Landslide types by material and movement (BRITISH GEOLOGICAL SURVEY). ....	20
Figure 2-15: Schematic composition of a landslide (VAN DEN EECKHAUT et al. 2012:32). ....	26
Figure 3-1: Study Area – Overview (OWN REPRESENTATION).....	28
Figure 3-2: Landslides with different ages in their morphological structure (LEOPOLD et al. 2015:18).....	30
Figure 4-1: ESP-Tool tool output example for temporary human settlements (DRĂGUȚ et al. 2010:864). .....	36
Figure 4-2: Objective function output example for Darjeeling (MARTHA et al. 2011:4934). ....	38
Figure 4-3: Concept of landslide detection (OWN REPRESENTATION).....	39
Figure 4-4: Concept of segmentation level 1 (OWN REPRESENTATION).....	42
Figure 4-5: Concept of segmentation level 2 (OWN REPRESENTATION).....	43
Figure 4-6: Concept of segmentation level 3 (OWN REPRESENTATION).....	45
Figure 4-7: Flow chart of segmentation function (OWN REPRESENTATION).....	51
Figure 4-8: Flow chart of objective function (OWN REPRESENTATION).....	52
Figure 4-9: Flow chart of getBoundingBox function (OWN REPRESENTATION).....	53
Figure 4-10: Flow chart of MainDirection function (OWN REPRESENTATION).....	54
Figure 4-11: Flow chart of RelationalObjectiveFunction (OWN REPRESENTATION).....	55
Figure 4-12: Flow chart of NeighborGrowing function (OWN REPRESENTATION).....	56
Figure 4-13: Flow chart of TextureFlow function (OWN REPRESENTATION) .....	57
Figure 5-1: Level 1 – Objective function (OWN REPRESENTATION).....	59
Figure 5-2: Segmentation level 1 (OWN REPRESENTATION) .....	62

Figure 5-3: Level 2 – Objective function (OWN REPRESENTATION) .....	64
Figure 5-4: Segmentation level 2 (OWN REPRESENTATION) .....	66
Figure 5-5: Level 2 – Objective function (OWN REPRESENTATION) .....	68
Figure 5-6: Segmentation level 3 (OWN REPRESENTATION) .....	76
Figure 5-7: Distribution of area in classification and inventory (OWN REPRESENTATION).....	78
Figure 6-1: False-positive landslides (OWN REPRESENTATION) .....	81
Figure 6-2: Examples of (not) detected landslides (OWN REPRESENTATION). ....	83
Figure 6-3: Difficulties with input parameters (OWN REPRESENTATION). .....	85
Figure 6-4: Creation of directional neighborhood (OWN REPRESENTATION). ....	89

## List of Tables

Table 2-1: Selection of land surface parameters and objects (adapted from WILSON & GALLANT (2000a:7ff.) and WILSON & BISHOP (2013:169ff.)) .....	7
Table 2-2: Shape metrics for objects of different shape (OWN REPRESENTATION). ....	16
Table 2-3: Further landslide causes and triggers (adapted from CRUDEN & VARNES 1996:70).....	22
Table 3-1: Summarized land cover of the study area (OWN REPRESENTATION). ....	27
Table 3-2: Landslide inventory statistics for the study area (OWN REPRESENTATION). ....	31
Table 4-1: morphometric characterization of different landslide parts (adapted from VAN DEN EECKHAUT et al. (2012:33)).....	40
Table 4-2: Overview of parameters used for landslide detection (OWN REPRESENTATION)....	41
Table 4-3: Explanation of flow chart symbols (OWN REPRESENTATION). ....	51
Table 5-1: Level 1 – k-means thresholds of land surface parameters (OWN REPRESENTATION). .....	60
Table 5-2: Level 1 – k-means thresholds of shape metrics (OWN REPRESENTATION).....	62
Table 5-3: Level 2 – k-means thresholds of land surface parameters and shape metrics (OWN REPRESENTATION). .....	65
Table 5-4: Level 3 – k-means thresholds of land surface parameters (OWN REPRESENTATION). .....	69
Table 5-5: Level 3 – k-means thresholds of <i>ClassNeighborFunction</i> on land surface parameters (OWN REPRESENTATION). .....	70
Table 5-6: Level 3 – k-means thresholds of textural features and shape metrics (OWN REPRESENTATION). .....	70
Table 5-7: Level 3 – k-means thresholds of <i>RelationalClassFunction</i> for class 10 (OWN REPRESENTATION). .....	72
Table 5-8: Accuracy assessment of landslide classification (OWN REPRESENTATION). ....	77
Table 5-9: Comparison between landslide inventory and classification (OWN REPRESENTATION). .....	78
Table 5-10: Landslide classification statistics for the study area and inventory (OWN REPRESENTATION). .....	79

## List of Acronyms

<b>2D</b>	two-dimensional
<b>ALS</b>	Airborne Laser Scanning
<b>CERL</b>	US Army Construction Engineering Research Lab
<b>CLC</b>	CORINE (Coordination of Information on the Environment) Land Cover
<b>CRED</b>	The Centre for Research on the Epidemiology of Disasters
<b>DEM</b>	digital elevation model
<b>DiDeM</b>	Programm für Digitale Gelände-Modellierung
<b>DTM</b>	digital terrain model
<b>DSM</b>	digital surface model
<b>EPSG</b>	European Petroleum Survey Group Geodesy
<b>ESP</b>	estimate scale parameter
<b>ESRI</b>	Environmental Systems Research Institute
<b>FOSS</b>	Free and Open Source Software
<b>FSF</b>	Free Software Foundation
<b>GEOBIA</b>	geographic object-based image analysis
<b>GIScience</b>	Geographic Information Science
<b>GK</b>	Gauss-Kruger
<b>GLCM</b>	grey level co-occurrence matrix
<b>GNSS</b>	Global Navigation Satellite Systems
<b>GNU GPL</b>	GNU General Public License
<b>GPS</b>	Global Positioning System
<b>GRASS</b>	Geographic Resources Analysis Support System
<b>IDL</b>	Interactive Data Language
<b>IDM</b>	Inverse difference moment
<b>IMU</b>	Inertial Measurement Unit
<b>INPE</b>	National Institute for Space Research
<b>IQR</b>	interquartile range
<b>LaDAR</b>	Laser Detection And Ranging
<b>laser</b>	Light Amplification by Stimulated Emission of Radiation
<b>LiDAR</b>	Light Detection And Ranging
<b>LV</b>	locale variance
<b>MATLAB</b>	matrix laboratory
<b>MGI</b>	Militärgeographisches Institut
<b>NH</b>	normalized height
<b>OBIA</b>	object-based image analysis
<b>OSGeo</b>	Open Source Geospatial Foundation
<b>OSI</b>	Open Source Initiative
<b>PCA</b>	principal component analysis
<b>ROC-LV</b>	rate of change of local variance

<b>RVT</b>	Relief Visualization Toolbox
<b>SADO</b>	System zur Analyse und Diskretisierung von Oberflächen
<b>SAGA</b>	System for Automated Geoscientific Analyses
<b>SARA</b>	System zur Automatischen Reliefanalyse
<b>SLIC</b>	simple linear iterative clustering
<b>SPRING</b>	Sistema de Processamento de Informações Georreferenciadas
<b>TauDEM</b>	Terrain Analysis Using Digital Elevation Models
<b>UNESCO</b>	United Nations Educational, Scientific and Cultural Organization
<b>USPO</b>	unsupervised segmentation parameter optimization
<b>VRM</b>	vector ruggedness measure
<b>WP/WLI</b>	Working Party on World Landslide Inventory

## 1 Introduction

Landslides are stunning phenomena, but they can also be natural disasters causing great damage to human beings or even their death. The author's interest in the subject "landslides" was already raised in his first lecture on physical geography where the German documentary "Gefahr aus den Bergen" (*engl.* danger from the mountains, Terra X, 2009) was shown. In this movie, scientists examine the linkage between rockslides, melting of alpine permafrost and climate change.

According to the study of JAKOB & LAMBERT (2009) an increase of antecedent and short-term precipitation due to climate change, can lead to an increase in landslide frequencies during the twenty-first-century (JAKOB & LAMBERT 2009:283). Therefore, landslides and their predictions in form of susceptibility maps are an important factor for decision-making in hazard prevention and spatial planning (LEOPOLD et al. 2015:5f., STEGER et al. 2017:1).

There are different approaches for susceptibility mapping. However, one fundamental source for modelling is the landslide inventory (PETSCHKO et al. 2016:857). The quality of the inventory has therefore a high impact on the predictive power of any statistical model used (PETSCHKO et al. 2016:857, STEGER et al. 2017:1). Nevertheless, the creation of inventories is a time-consuming process and depends on the subjective assessment of experts (PETSCHKO et al. 2016:866, VAN DEN EECKHAUT et al. 2007:761ff.). To support the creation of inventories, multiple approaches have emerged to semi-, or full automatic detect landslides by using high-resolution data in the last decade. Especially, the use of high-resolution digital terrain models (DTM) enables the possibility to automatically identify landslides even under forest area, where images from passive sensors are limited (VAN DEN EECKHAUT et al. 2012). Here, an object-oriented image analysis (OBIA) is usually preferred in comparison to a pixel-based approach (MARTHA et al. 2011, STUMPF & KERLE 2011, VAN DEN EECKHAUT et al. 2012). However, these OBIA approaches are mainly based on commercial software such as Definiens eCognition, and fewer investigations have been made yet for using open-source products for the landslide detection.

The thesis project has the objective to automatically detect landslides by an object-oriented approach using only high-resolution LiDAR DTM derivatives and open-source software. An expert-based landslide inventory has already been created for the study area – in the province of Burgenland, Austria. This way it is possible to compare the classification with the inventory in order to 1. assess the accuracy of the classification, and 2. show the potential for the support of the creation of an inventory.

Chapter 2 to chapter 7 are structured as the following: Chapter 2 focuses on the theoretical background of LiDAR, landslides, and the parameters used for the classification. Moreover, the state of the art and research questions are presented. Chapter 3 introduces the study area of this thesis, as well as the data on which this work is based: the DTM and the landslide inventory. In chapter 4 the object-oriented approach for the detection of landslides is presented, and is divided into three parts: In the first part, the theory, workflow, and accuracy assessment of the classification are explained. The second part has its focus on the used open-source products. Chapter 4 ends with an overview of the implemented functions created for the detection algorithm, written in the development environment R and packed up in the *Lslide* package. Chapter 5 provides the results of the segmentation processes, as well as the achieved accuracies including a comparison with the inventory. The results are embedded into the actual state of the art in chapter 6, and the limitations as well as the possibilities of the classification are discussed. Chapter 7 summarizes the main subjects and results of this thesis, and provides further questions for research.

The programming code can be accessed by the following links and calls (*last access: 16-04-2017*):

- *Lslide* package: <https://github.com/raff-k/Lslide>,
- *Lslide* package in R: `devtools::install_github("raff-k/Lslide")`,
- scripts for the detection: <https://github.com/raff-k/Lslide-thesis>.

## 2 Theoretical Background and Research Question

### 2.1 Fundamentals

#### 2.1.1 LiDAR

In this chapter principal aspects of the subject LiDAR are explained. CAMPBELL & WYNNE (2011) give a general overview about LiDAR, while WEHR & LOHR (1999) provide more physi-technical explanations.

##### 2.1.1.1 Definition

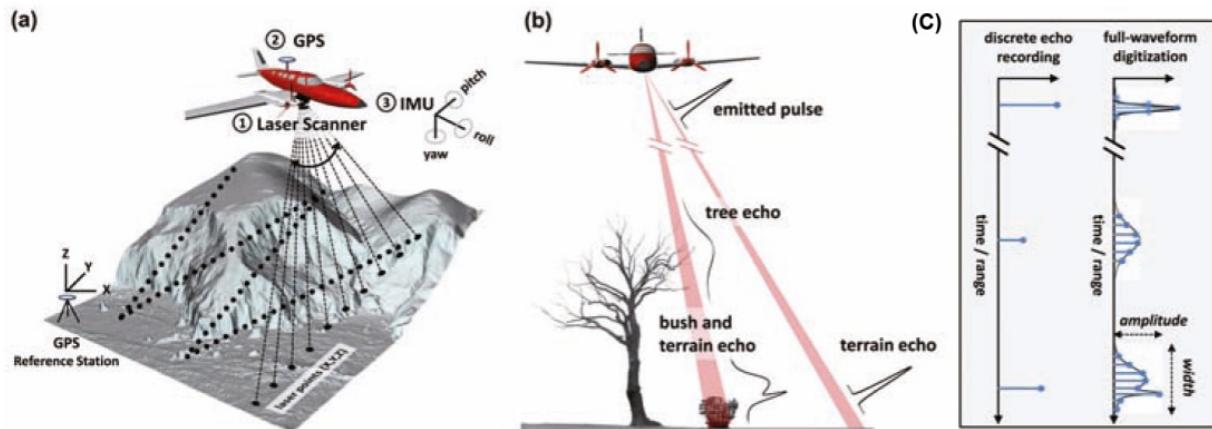
LiDAR stands for the acronym “light detection and ranging” and is a remote sensing system with active sensors. That means they transmit their own energy to a target surface, receive the backscattered energy and calculate an image of the Earth’s surface by measuring the distance between them (CAMPBELL & WYNNE 2011:243). In literature, the LiDAR synonyms LaDAR – for “laser detection and ranging”, and laser scanning, are frequently found. Whereas the former term emphasizes that laser light is used for distance measurements, the latter term accentuates an area-wise acquisition of the target surface with a lot of single measurements (HÖFLE & RUTZINGER 2011:2). Most LiDAR systems emit visible light with wavelength ranging between 532-1,640 nm (CAMPBELL & WYNNE 2011:245f.).

There are four principal types of LiDAR applications: spaceborne, airborne, mobile and terrestrial (HÖFLE & RUTZINGER 2011:3). In this thesis, only airborne LiDAR or Airborne Laser Scanning (ALS) systems were of interest, and are therefore generally termed as LiDAR continuing from here. In ALS systems, airborne sensors are generally mounted on manned or unmanned flying vehicles like aircrafts, helicopters or drones (HÖFLE & RUTZINGER 2011:3).

LiDAR’s are based on lasers – an acronym for “light amplification by stimulated emission of radiation”. In the late 1950s lasers were invented. They have the capability to emit a coherent beam within a narrow optical spectrum with usually 10 nm bandwidth (CAMPBELL & WYNNE 2011:243f., WEHR & LOHR 1999:74). In the first airborne applications profiling lasers, also known as airborne laser altimeters, were used for the creation of elevation profiles by providing a high density of observations. In the late 1980s the development of precision scanning LiDAR systems emerged and were able to produce entire images of the Earth’s surface (CAMPBELL & WYNNE 2011:245).

##### 2.1.1.2 Mode of Operation

The airborne LiDAR system is a multi-sensor measurement system which consists of three principal components (HÖFLE & RUTZINGER 2011:3): (1) laser scanner, (2) Global Positioning System (GPS), and (3) an Inertial Measurement Unit (IMU) (Figure 2-1).



a: system components of a typical ALS. b: emitting and detection of signals. c: range measurement principles.

Figure 2-1: Concept of Airborne LiDAR (HÖFLE & RUTZINGER 2011:4).

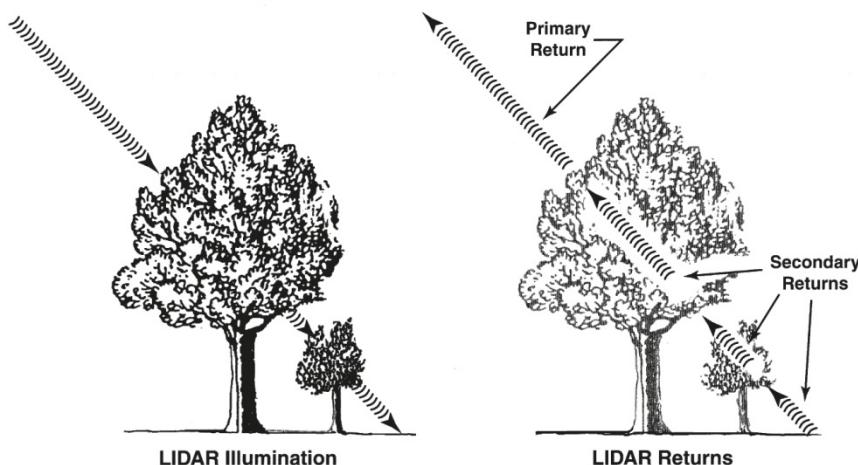
(1) The typical laser scanner consists of several key elements (WEHR & LOHR 1999:68): laser ranging unit, opto-mechanical scanner, and control and processing unit. For more details refer to WEHR & LOHR (1999). (2) The GPS determines the geographical location of an aircraft by reference stations or Global Navigation Satellite Systems (GNSS) (HÖFLE & RUTZINGER 2011:3). (3) The IMU controls the orientation of an aircraft (yaw, pitch, and roll) relative to the platform-fixed coordinate system (CAMPBELL & WYNNE 2011:245, WEHR & LOHR 1999:79).

To calculate the exact 3-D position of a measurement the LiDAR system requires an exact time synchronization of the three above mentioned components (WEHR & LOHR 1999:78).

Depending on the design of the LiDAR instrument, there are two major range measurement principles (HÖFLE & RUTZINGER 2011:4): discrete return recording system and full-waveform digitization system. In literature, there are different terms used for the backscatter of an emitted beam depending on the context (HÖFLE & RUTZINGER 2011:4). These terms are “return”, “echo”, “reflection” or “pulse”, which are also used as synonyms in this thesis.

The discrete return recording system measures the travelling time of an emitted light pulse until its detection at the receiver (WEHR & LOHR 1999:70). Beside distance measurements, the most commercial sensors are also able to store the strength of a pulse reflection, known as signal intensity (HÖFLE & RUTZINGER 2011:4). While at the end of the 1990s pulse lasers were most popular (WEHR & LOHR 1999:70), nowadays, the new generation of commercial ALS use the full-waveform ranging principle (HÖFLE & RUTZINGER 2011:4f.).

The full-waveform digitization system is based on a continuous signal transmitted by the laser. Here, the travelling time of a signal is calculated by the phase difference between received and transmitted signal (WEHR & LOHR 1999:70f.). This method allows the detection of distinct returns of a single laser shot, the derivation of the signal amplitude, and the width of each return (HÖFLE & RUTZINGER 2011:5).



Left: emitted pulse. Right: different returns due to tree (stem, branches, ground).

Figure 2-2: Illustration of LiDAR returns at a tree (CAMPBELL & WYNNE 2011:251).

Different kinds of recorded returns from the target surface exist (Figure 2-2) (CAMPBELL & WYNNE 2011:250, HÖFLE & RUTZINGER 2011:4): Primary returns, also referred to as first return, indicate the first objects a LiDAR pulse encounters. This could be the upper part of a vegetation canopy for example. Backscatters from lower parts of a surface object, for example from leaves, branches, or the ground surface itself, are known as secondary, or partial returns. If the secondary return is the last LiDAR pulse received from an object, it is referred to as last return.

Within the maximum distances of about 800-1000 m the measurement accuracy of recent laser instruments is about 1.5 mm (MANETTI & STEINMANN 2007:5). The accuracy of the absolute xyz-position of a recorded reflection depends on the quality of the multi-sensor measurement system of the LiDAR, but it is stated to be of horizontal accuracy in the range of 20-30 cm and of vertical accuracy in the range of 15-20 cm (CAMPBELL & WYNNE 2011:250). Nowadays, sensor systems are able to detect high point densities (> 10 points per m<sup>2</sup>) providing a very detailed description of the terrain surface (HÖFLE & RUTZINGER 2011:5). However, the quality of the end-user product such as DTMs, digital surface models (DSM), or city models is mainly dependent on the filtering and interpolation algorithms of the (mostly commercial) post-processing software (WEHR & LOHR 1999:80).

### 2.1.1.3 Advantages, Limitations and Applications

As an active sensor system LiDARs are independent of solar illumination. Due to their multi-measurement system they have also the potential to assess, besides the brightness of the backscatter, also its angular position, changes in frequency, and the timing of reflected pulses (CAMPBELL & WYNNE 2011:243). Additionally, by recent sensor systems with high point densities, there is the possibility to penetrate vegetation by “seeing” through gaps (HÖFLE & RUTZINGER 2011:5). Thus, highly accurate and detailed information of terrain, vegetation

features, or cities can be extracted from LiDAR data (Figure 2-3), leading to new possibilities in a broad range of environmental applications (CAMPBELL & WYNNE 2011:252, 2011:254).

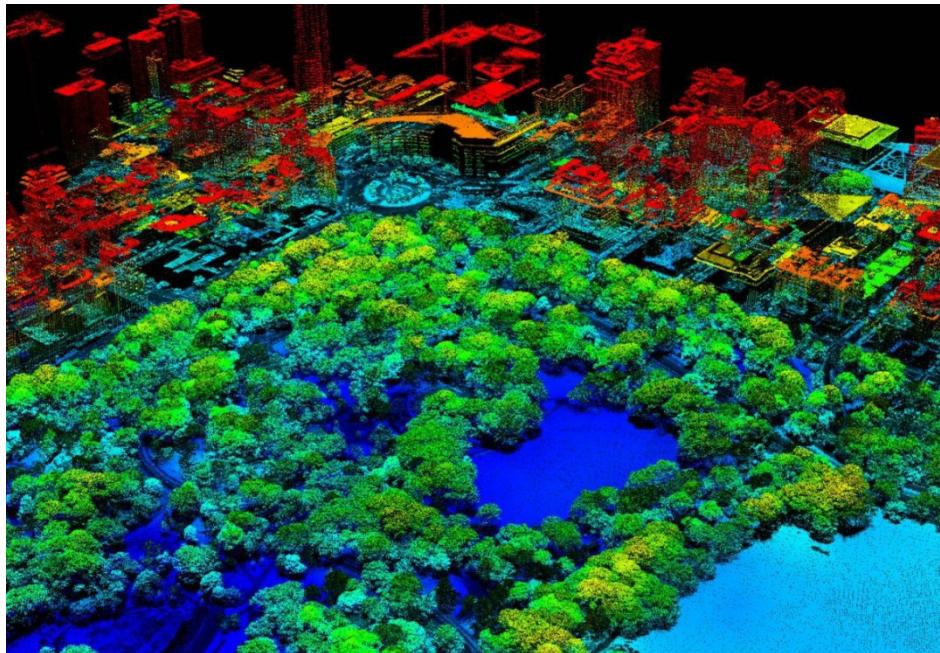


Figure 2-3: LiDAR point cloud from Central Park in New York City (O'NEIL-DUNNE).

However, HÖFLE & RUTZINGER (2011) mention that the acquisition costs a lot and the big amount of data for large regions are still challenging (HÖFLE & RUTZINGER 2011:5). Moreover, in near future a global coverage is not expected to be achieved. Projects with repetitive data collection are mainly focused on specific regional or local study areas (HÖFLE & RUTZINGER 2011:5). In addition, JABOYEDOFF et al. (2012) state that under certain conditions such as bad weather, poorly reflecting or very rough surfaces, or very bright ambient conditions, the instrument accuracy is diminished (JABOYEDOFF et al. 2012:8). Moreover, high resolution can also lead to more noise or artefacts in data (MINER et al. 2010:7).

Some important applications of airborne LiDAR are (WEHR & LOHR 1999:81):

- mapping of infrastructures (roads, railway tracks, electrical transmission lines),
- generation of high-resolution terrain surface models (DTM, DSM),
- support in hydrographic surveys,
- survey of sensible eco-systems (coastal areas, forest areas),
- or rapid mapping and damage assessment after natural disasters.

### 2.1.2 Geomorphometry and Land Surface Parameter

In this chapter the term “geomorphometry” and its most important derivatives – land surface parameters, are introduced. However, only land surface parameters that are used in this thesis are explained in detail. The derivation of these parameters is based on the DTM of the study area which is presented in chapter 3.2. Therefore, different settings for the derivation mentioned in the following are with respect to this DTM.

A general overview can be found in WILSON & BISHOP (2013) and OLAYA (2009).

### 2.1.2.1 Definition

Geomorphometry is the quantitative characterization of the land surface - also known as terrain analysis, quantitative geomorphology, or digital terrain modelling (PIKE 2000:1f.). It takes an important role in the study of Earth's topography, especially for describing surface processes and landforms. Geomorphometry emerged as an interdisciplinary field from mathematics, Earth science – especially geomorphology and civil as well as military engineering. Nowadays, geomorphometry evolves by the influence of computer sciences (PIKE et al. 2009:12). Since the end of the 1980s computing power increased enormously, accompanied by new processing algorithm and the possibility of acquiring an increasing amount of higher-resolution data (PIKE et al. 2009:22). Information on elevation, normally in digital form, serve here as the fundamental input for the analysis (PIKE et al. 2009:3, WILSON & BISHOP 2013:163).

Modern geomorphometry has the operational aim to extract land surface parameters and objects from digital terrain models (DTM) (Table 2-1). Land surface parameters are descriptive measures of the continuous Earth surface and can be subdivided into primary and secondary parameters (PIKE et al. 2009:6, WILSON & GALLANT 2000a:6). Primary parameters are directional derivatives of a topographic surface such as slope or aspect. Secondary parameters are calculated by two or more primary parameters and usually describe spatial variability in landscape processes by physically based or empirically derived indices. The topographic wetness index or the stream-power index are examples for this kind of parameters (GALLANT & WILSON 2000:51ff., WILSON & GALLANT 2000a:6, 2000b:87ff.). Land surface objects are discrete spatial features representing segments of the landscape as spatial entities. They are mostly computed as vectors consisting of points, lines, or polygons like a watershed line or drainage network for example (PIKE et al. 2009:6, WILSON & BISHOP 2013:163).

**Table 2-1: Selection of land surface parameters and objects (adapted from WILSON & GALLANT (2000a:7ff.) and WILSON & BISHOP (2013:169ff.))**

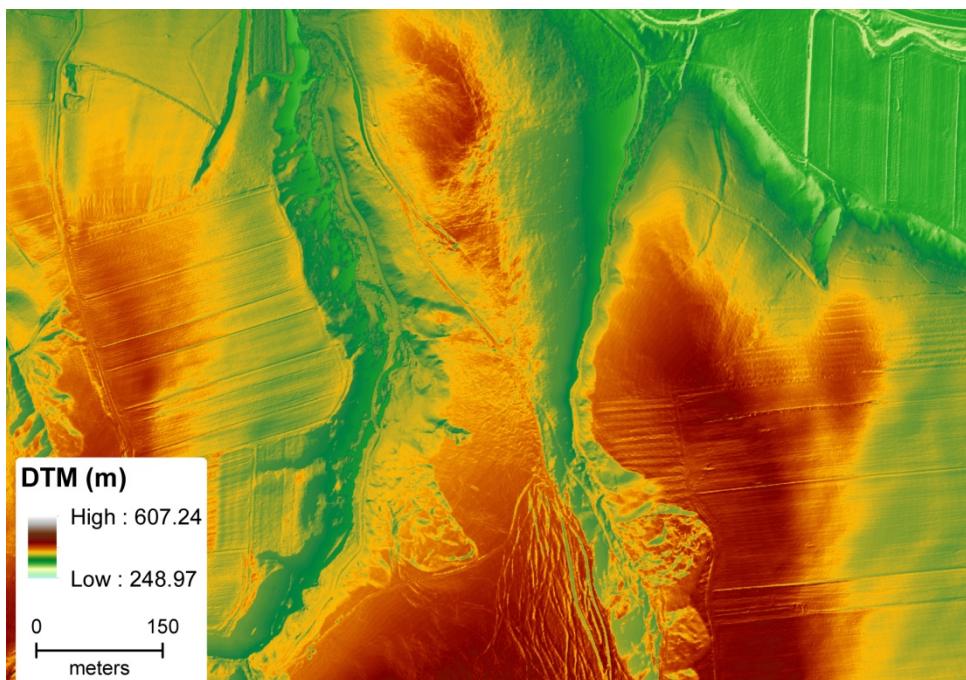
land surface parameter	land surface object
<p><i>Primary Parameter</i></p> <ul style="list-style-type: none"> <li>▪ slope (upslope, dispersal, catchment)</li> <li>▪ curvature (profile, plan, tangential)</li> <li>▪ aspect</li> <li>▪ catchment or dispersal area</li> <li>▪ flow path, upslope, dispersal or catchment length</li> <li>▪ surface roughness</li> </ul> <p><i>Secondary Parameter</i></p> <ul style="list-style-type: none"> <li>▪ topographic wetness indices</li> <li>▪ stream-power indices (sediment transport capacity index, erosive power of flowing water)</li> <li>▪ radiation indices (long-wave irradiance, net radiation)</li> <li>▪ temperature indices</li> </ul>	<p><i>land surface object</i></p> <ul style="list-style-type: none"> <li>▪ watershed line</li> <li>▪ cirque</li> <li>▪ alluvial fan</li> <li>▪ drainage network</li> <li>▪ hillslope units (summits, shoulders, back, foot and toe slopes)</li> </ul>

### 2.1.2.2 DTM

In the frame of this thesis a digital terrain model (DTM) is defined as a bare-earth elevation model of land surfaces, which means that in elevation no land surface forms, such as buildings or trees, are included. Commonly, the DTM is in form of a squared-grid representing the topography (Figure 2-4).

This definition is synonymous to the definition of a digital elevation model (DEM) in WILSON & BISHOP (2013:162ff.) and in PIKE et al. (2009:4). In some literature also the term digital relief model can be found expressing a DTM (PIKE et al. 2009:5).

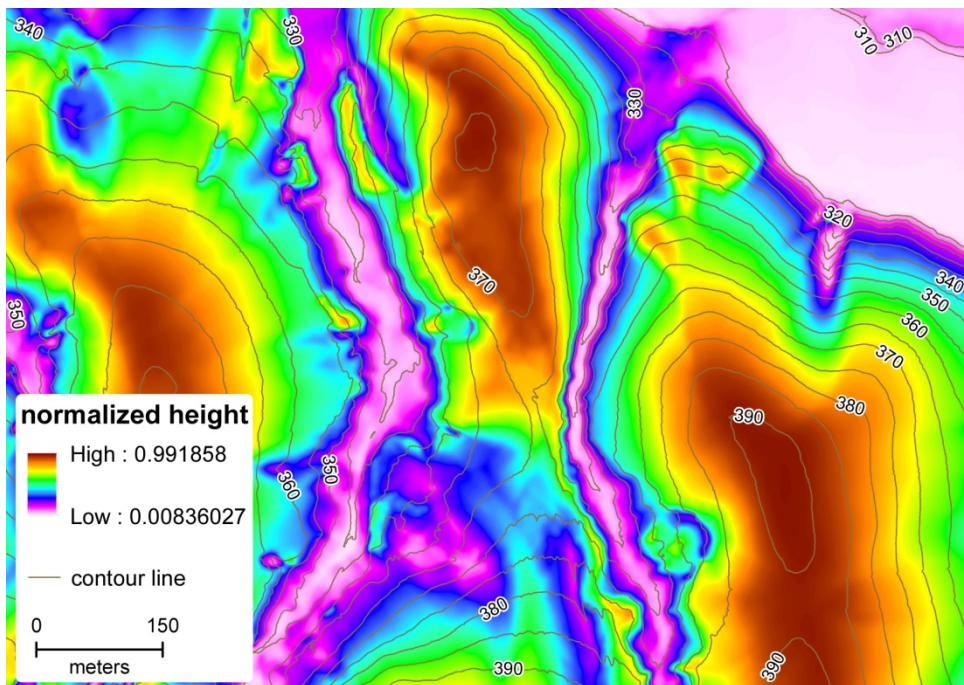
In contrast, WILSON & BISHOP (2013) define a DTM as workflow from altitude samples to the creation of surface models with the correction of errors and artifacts to the final production of a DEM (WILSON & BISHOP 2013:163f.).



Anonymized extract (s. 3.2).

Figure 2-4: Land surface parameter: digital terrain model (OWN REPRESENTATION).

Besides the DTM, also the normalized height (NH) according to DIETRICH & BÖHNER (2008) was used for this thesis (Figure 2-5). The NH is based on the DTM and provides information on the location of a terrain point to its local environment. The values range from 0 for at the bottom of a slope to 1 for at the top of a slope (DIETRICH & BÖHNER 2008:42). Due to the high resolution of the DTM (s. 3.2), the output of the NH was very noisy. Therefore, for this study the DTM was resampled from 1m to 3 m resolution by a bicubic spline interpolation. Moreover, the NH was created using smoothing criteria and a small weighting of the catchment area ( $w = 5$ ,  $t = 2$ ,  $e = 2$  ).



Anonymized extract (s. 3.2).

Figure 2-5: Land surface parameter: normalized height (OWN REPRESENTATION).

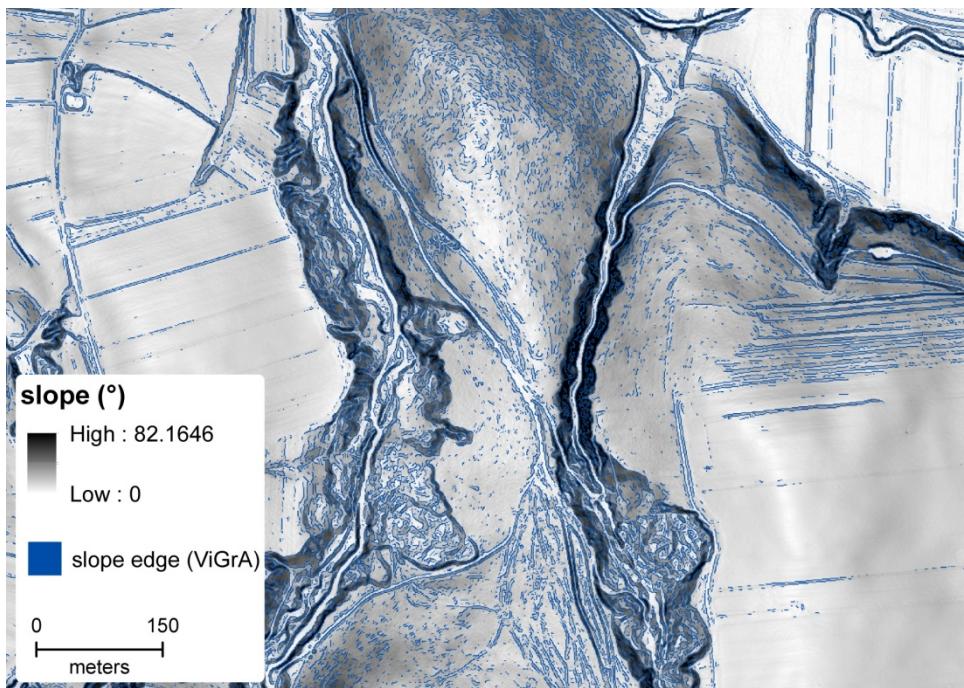
### 2.1.2.3 Slope and Curvature

The slope and curvature of a surface are based on the first and second derivative of a DTM, respectively (OLAYA 2009). There are various algorithms to derive those parameters. However, only the methods used in this thesis are mentioned.

The slope expresses the maximal rate of change of elevation values inside of a defined window size, and was here calculated by using the method of ZEVENBERGEN & THORNE (1987) (Figure 2-6). The method uses a  $3 \times 3$  cell window size, and interpolates the slope by using Lagrange polynomials (ZEVENBERGEN & THORNE 1987:49).

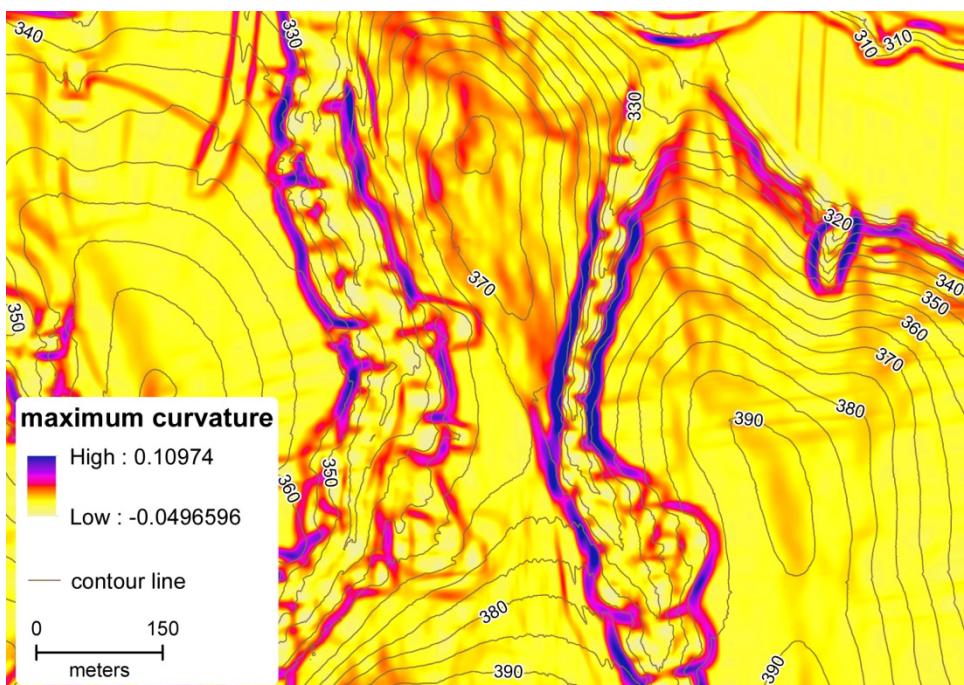
The curvature represents the concavity and convexity of the land surface. There are different types of curvature such as profile (or vertical), tangential (or horizontal), plan, cross-sectional, longitudinal, maximum or minimum curvature that are explained in detail in WOOD (1996) or OLAYA (2009). The most standardized methods calculate curvature using a  $3 \times 3$  cell window. However, with the modification of WOOD (1996), it is possible to compute curvature (and other parameters) across scales using different window sizes.

Therefore, for this study the maximum curvature was calculated using the method of WOOD (1996) (Figure 2-7). The maximum curvature expresses the surface convexity and is solely based on surface geometry. Thus, it has the advantage to be independent of the slope (TAROLLI et al. 2012:71, WOOD 1996:86).



Anonymized extract (s. 3.2).

Figure 2-6: Land surface parameter: slope (OWN REPRESENTATION).



Anonymized extract (s. 3.2).

Figure 2-7: Land surface parameter: maximum curvature (OWN REPRESENTATION).

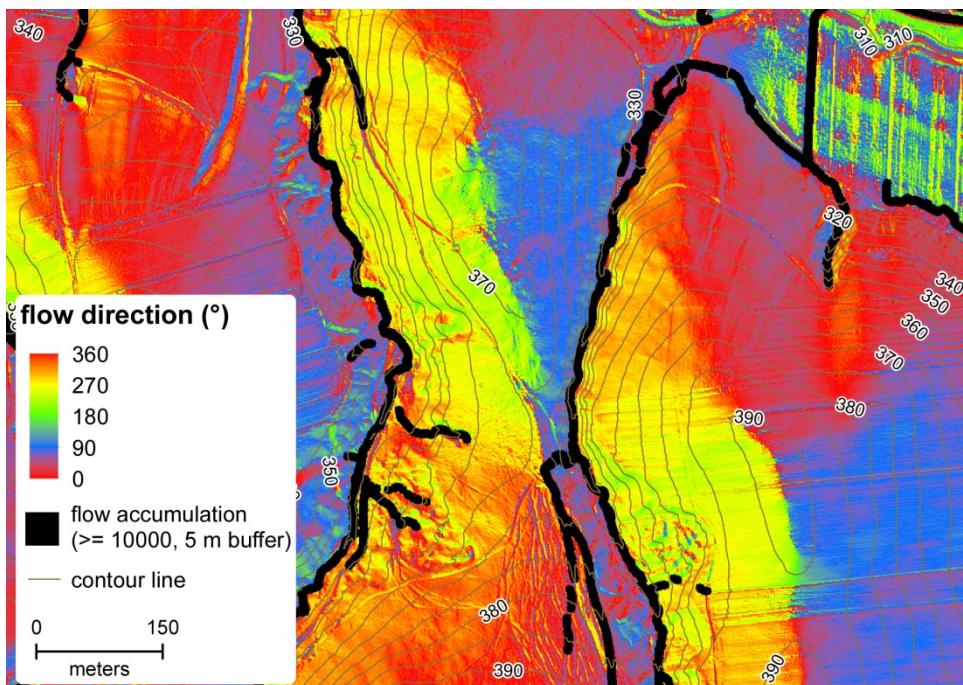
#### 2.1.2.4 D-Infinity Flow Direction and Accumulation

The flow direction is an important component in hydrological modelling and serves as basis for other procedures such as flow accumulation or watershed delineation (GRUBER & PECKHAM 2009:172). The flow direction provides information about the potential direction of water drain of a cell. GRUBER & PECKHAM (2009) introduced the most commonly used algorithms, as well as their advantages and disadvantages. The common approaches are:

method of the steepest descent (D8), multiple flow direction (MFD), D-Infinity ( $D^\infty$ ), DEMON, and Mass-Flux Method (MFM) (GRUBER & PECKHAM 2009:176ff.).

Every method has its benefits and drawbacks. However, in comparison to the other flow algorithms the  $D^\infty$  algorithm, developed by TARBOTON (1997), a good compromise: It overcomes the disadvantages of the over-dispersion in low elevations in the case of MFD and of the grid bias in high elevations in the case of the D8 (GRUBER & PECKHAM 2009:178). Therefore, the  $D^\infty$  algorithm was used in this study (Figure 2-8).

The flow accumulation, sometimes also referred to contributing -, basin -, or upslope area, describes a planar area with the collection of drainage from other pixels (GRUBER & PECKHAM 2009:182). The information from where the inflow of a cell derives from and where the drainage is leading to is provided by the flow direction. Therefore, the spatial pattern of a flow accumulation is dependent on the method used for calculating the flow direction (GRUBER & PECKHAM 2009:183).



Anonymized extract (s. 3.2).

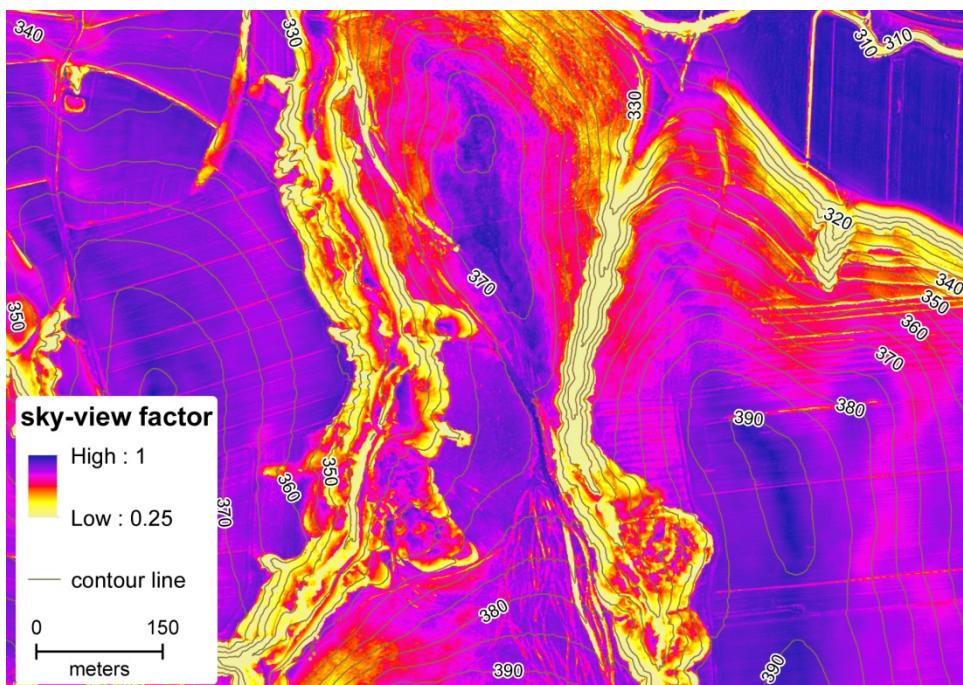
Figure 2-8: Land surface parameter: D-Infinity flow direction and accumulation (own representation).

### 2.1.2.5 Sky-View Factor

Relief shading is often used to visually support the recognition of specific relief characteristics. According to ZAKŠEK et al. (2011) in most GIS software the analytic hill-shading method of YOËLI (1965) is implemented (ZAKŠEK et al. 2011:400). With this method the terrain is illuminated by direct light from a fictive light source. However, linear structures located parallel to the light beam are difficult to recognize. Furthermore, there is a loss of details in dark shades and brightly lit areas, when the incidence angle becomes greater than 90°, or close to zero, respectively (ZAKŠEK et al. 2011:400).

To overcome these limitations ZAKŠEK et al. (2011) developed the sky-view factor (Figure 2-9). Their illumination method is based on diffuse illumination with a fictive light source from a celestial hemisphere (ZAKŠEK et al. 2011:401f.). With the method, it is even possible to highlight small scale features by using a small search window, such as for example small ridges in a relatively flat relief (ZAKŠEK et al. 2011:412). The value of the sky-view factor ranges from 0 to 1, with 1 for the entire visibility of the sky (planes, peaks), and 0 for no visibility of the sky (deep sinks, lower part of a deep valley) (ZAKŠEK et al. 2011:402).

In this study, the sky-view factor was calculated using the Relief Visualization Toolbox (s. 4.2.4) according to ZAKŠEK et al. (2011).



Number of directions: 16. Search radius: 20 m. Anonymized extract (s. 3.2).

Figure 2-9: Land surface parameter: maximum curvature (OWN REPRESENTATION).

### 2.1.2.6 Openness

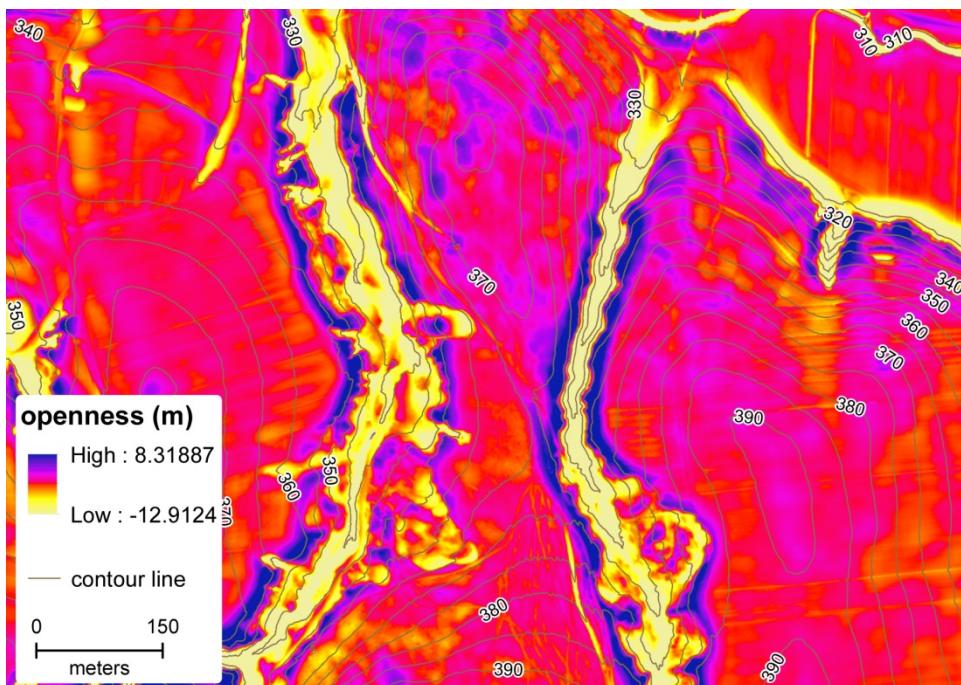
The term “openness” in geomorphometry is often related to the image-processing technique introduced by YOKOYAMA et al. (2002). Their algorithm calculates the topographical dominance or enclosure for any position of the land surface (YOKOYAMA et al. 2002:263). Here, a distinction is made between positive and negative openness. The former highlights convex landforms and can show areas with extreme relief. The latter underlines concave landforms and is capable to show the scale-dependent hydrological network for example (WILSON & BISHOP 2013:171, YOKOYAMA et al. 2002:263).

However, for this study the term openness is adapted from VAN DEN EECKHAUT et al. (2012) (Figure 2-10). They define openness as difference between the original DTM and a smoothed DTM of a suitable window size (VAN DEN EECKHAUT et al. 2012:34). According to

VAN DEN EECKHAUT et al. (2012) the window size ( $k$ ) was set to 51. Therefore, the openness was calculated as follows:

$$\text{openness} = DTM - DTM_{k_{51}}$$

Negative values can indicate depressions or valleys, while positive values indicate peaks.

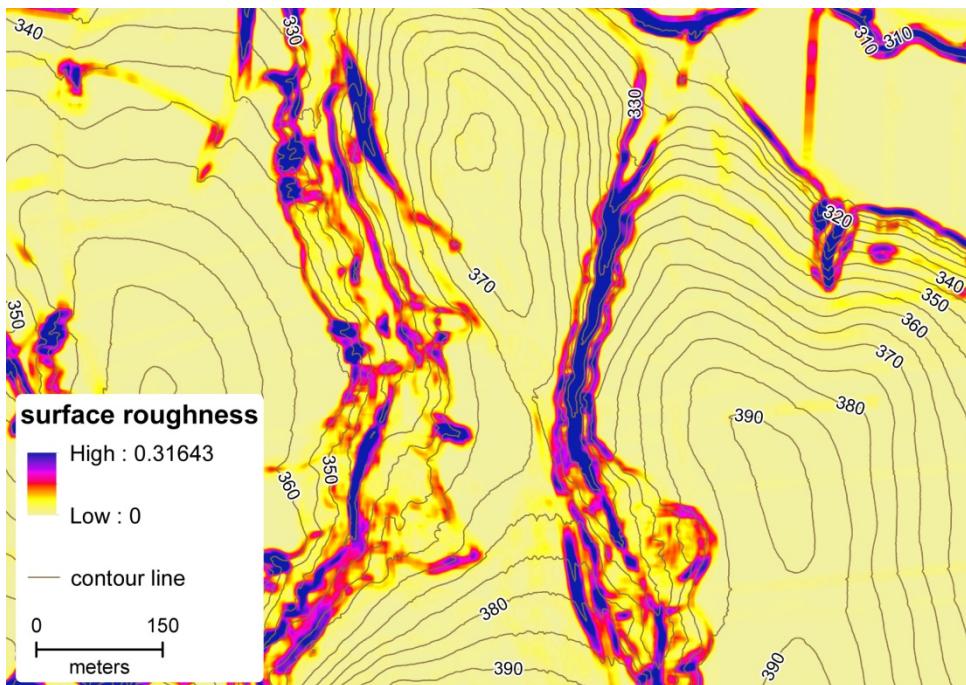


Anonymized extract (s. 3.2).

Figure 2-10: Land surface parameter: openness (OWN REPRESENTATION).

### 2.1.2.7 Surface Roughness

The surface roughness – or in literature also found as terrain ruggedness or terrain roughness, is a measurement for the undulation or complexity of the terrain (OLAYA 2009:158). According to OLAYA (2009), there is still no agreement how to calculate the surface roughness, and thus, a wide range of algorithms exist (OLAYA 2009:158). Nonetheless, the surface roughness is an important parameter for a number of scientific disciplines such as geomorphology, wind engineering, or ecology (SAPPINGTON et al. 2007:1419). The surface roughness can be easily computed by calculating the standard deviation of a slope for a given windows size (OLAYA 2009:158). However, SAPPINGTON et al. (2007) stated that an ideal measurement is a multivariate representation of the surface and, thus, propose to include the variability of slope and aspect for the calculation (SAPPINGTON et al. 2007:1419). They developed the so-called vector ruggedness measure (VRM), which is used in this thesis (Figure 2-11). The value of the VRM ranges from 0 for flat to 1 for most rugged areas (SAPPINGTON et al. 2007:1421).



Moving-window size: 15 m. Anonymized extract (s. 3.2).

Figure 2-11: Land surface parameter: surface roughness (OWN REPRESENTATION).

### 2.1.3 Shape Metrics

BOYCE & CLARK (1964) already mentioned the importance of shapes or forms as descriptive tool in earth-science with four main applications, namely, for describing urban forms, trade areas, political areas, and physical features (BOYCE & CLARK 1964:561). Meanwhile, a own discipline, celled landscape ecology, emerged and is concerned with the spatial configurations of the land surface and its interconnections between human and ecological systems (LEITÃO et al. 2006:1ff.). In remote sensing, with the increasingly upcoming object oriented image analysis (s. 4.1.1), spectral data are frequently combined with shape information. They are particularly useful when image objects are spectrally similar but physically differ in their representation (VAN DER WERFF & VAN DER MEER 2008:251). Basic shape parameters were used in this study and are therefore briefly explained below.

Typical shape metrics, that can be computed with common desktop GIS software such as SAGA GIS (s. 4.2.3), are the following:

- area ( $A$ ),
- perimeter ( $P$ ),
- interior edge ratio ( $P/A$ ),
- shape index ( $P/(2\sqrt{\pi A})$ ),
- inner maximum intervertex distance ( $D$ ) and its ratio to area ( $D/A$ ),
- ratio of perimeter and maximum distance to the square root of area ( $P/\sqrt{A}$ ,  $D/\sqrt{A}$ ).

However, within the frame of the thesis, these shape measurements are extended by the measurement of compactness, convexity, main direction and length-width-ratio of a shape object or polygon, respectively.

The compactness is calculated according to the formula in VAN DER WERFF & VAN DER MEER (2008) as follows:

$$\text{Compactness} = \frac{4 \pi \text{Area}}{\text{Perimeter}^2}$$

Therefore, the most compact object is a circle and has the maximum value of 1 (VAN DER WERFF & VAN DER MEER 2008:252).

For the calculation of the convexity, as first step, a convex hull for every object is computed. Then, the convexity can be measured by using the definition by ZUNIC & ROSIN (2004):

$$\text{Convexity} = \frac{\text{Area}}{\text{Area}_{\text{Convex Hull}}}$$

The values of this metric range from 0 to 1, with 1 representing an object with the highest convexity (ZUNIC & ROSIN 2004:923).

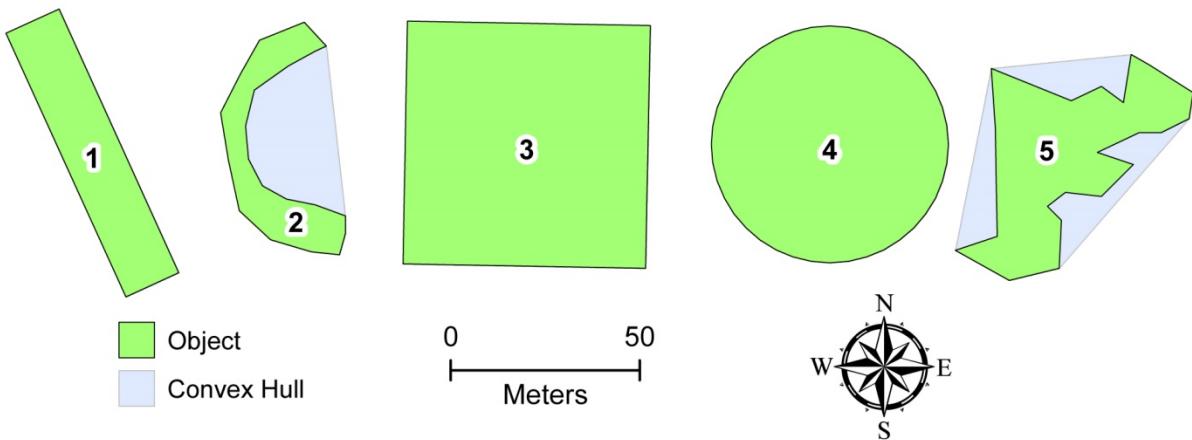
The main direction provides information on the orientation of an object in XY-space. For the computation of the orientation a principal component analysis (PCA) is applied on the X- and Y-coordinates of every object according to YI & MARSHALL (2000). The vectors of each object are extracted by combining eigenvalues and coordinates of the corresponding principal component. The main direction represents simply the direction of the vector with the largest eigenvalues, or biggest length, respectively, and is measured in degrees ranging from 0° to 360°. However, without a fix starting point the direction of a vector seems always to be ambiguous. Therefore, besides the main direction, its inversion ( $\pm 180^\circ$ ) is also calculated (s. Figure 4-10).

The computation of the length-width-ratio is basically identical to the one of the main direction above. However, after applying the PCA on an object and retrieving their vectors, the vectors are set to the following ratio:

$$\text{Length Width Ratio} = \frac{\text{longer Vector}}{\text{smaller Vector}}$$

The smallest value is 1 and represents an object with a balanced length to width ratio (s. Figure 4-10).

The following Figure 2-12 shows objects with different shapes, and in the related Table 2-2 the computation of the shape metrics are displayed.



N: 0|360°, E: 90°, S: 180°, W: 270°.

Figure 2-12: Objects of different shape (own representation).

Table 2-2: Shape metrics for objects of different shape (own representation).

shape metrics	1	2	3	4	5
area ( $m^2$ )	1184.88	684.61	4134.81	3077.27	1595.83
perimeter (m)	184.36	182.81	257.21	196.85	245.98
interior edge ratio	0.1556	0.2670	0.0622	0.0640	0.1541
perimeter-sqrt-area-ratio	5.36	6.99	4.00	3.55	6.16
shape index	1.51	1.97	1.13	1.00	1.74
max. distance (m)	78.28	62.34	90.94	62.72	75.58
max. distance-area-ratio	0.0661	0.0911	0.0220	0.0204	0.0474
max. distance-sqrt-area-ratio	2.27	2.38	1.41	1.13	1.89
compactness	0.44	0.26	0.79	1.00	0.33
convexity	1.00	0.46	1.00	1.00	0.66
main direction (°)	153.59	173.90	136.06	180.00	42.74
main direction inverse (°)	333.59	353.90	316.06	360.00	222.74
length-width-ratio	25.47	4.03	1.40	1.05	3.19

## 2.1.4 Textures

Besides spectral and contextual information on image objects, texture is another fundamental in pattern recognition. The texture provides important information on the structural composition of the surface and their relationship to its surroundings (HARALICK et al. 1973:610f.). In this thesis texture is defined and calculated based on the work of HARALICK et al. (1973). In their concept, texture and (gray) tones are in a dependent relationship (HARALICK et al. 1973:611). Moreover, it is assumed that different textural information originates from the overall spatial relationship that gray tones have to each other. This spatial relationship can be expressed by a set of gray-tone spatial-dependence matrices, or also known as gray level

co-occurrence matrix (GLCM), for various angular relationships ( $0^\circ$ : N–S,  $45^\circ$ : NO–SW,  $90^\circ$ : E–W,  $135^\circ$ : SW–SE) and distances (HARALICK et al. 1973:612). HARALICK et al. (1973) proposed 14 different textural features that can be expressed by its range or mean for the four directions (HARALICK et al. 1973:618f.). The rotation-invariance of a GLCM texture can be calculated by either taking the mean or minimum value of all four directions, or by summing up the four orientations before calculating the texture (STUMPF & KERLE 2011:2569).

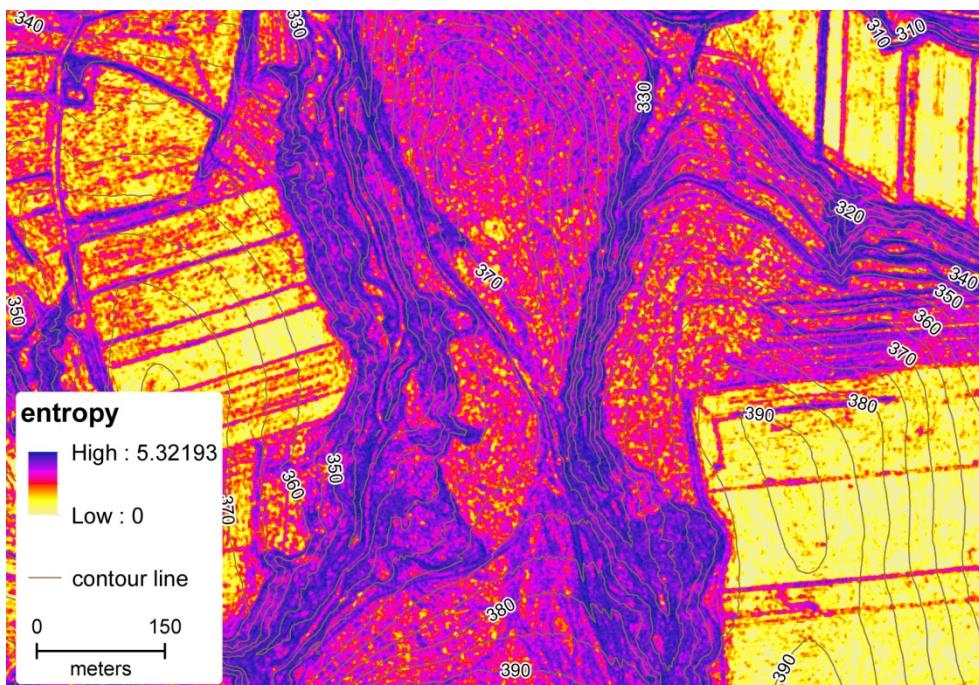
In common desktop GIS such as GRASS GIS (s. 4.2.2), the GLCM derivatives are computed pixel-wise by a moving window approach. However, in object-oriented analysis it is desirable to calculate the texture for every object. Fortunately, the open-source *EBImage*-package in R (s. 4.2.1) provides functionalities to calculate the mean rotational-invariance of all textures of HARALICK et al. (1973) for every object. Therefore, this function was used in this thesis (s. 4.3.7).

STUMPF & KERLE (2011) stated that landslide areas often have downslope-directed texture patterns (STUMPF & KERLE 2011:2569). Thus, the application of rotational-invariance measurements is not suitable. Therefore, STUMPF & KERLE (2011) suggested to pixel-wise derive directional GLCM measurements in dependence on the hydrological flow direction ( $\text{GLCM}_{\text{Flow}}$ ) and on the direction perpendicular to the flow direction ( $\text{GLCM}_{\text{FlowPer}}$ ). That means, if a cell has a flow direction value of  $45^\circ$ , for the  $\text{GLCM}_{\text{Flow}}$  output the value of the directional texture of  $45^\circ$  and for the  $\text{GLCM}_{\text{FlowPer}}$  output the value of the directional texture of  $135^\circ$  is used, respectively (STUMPF & KERLE 2011:2569). In their study, STUMPF & KERLE (2011) used then textures derived by the  $\text{GLCM}_{\text{Flow}}$  as well as by the ratio of  $\text{GLCM}_{\text{Flow}}$  to  $\text{GLCM}_{\text{FlowPer}}$  ( $\text{GLCM}_{\text{Ratio}}$ ). Besides the *EBImage* function, this method was thence also implemented in this study (s. 4.3.6).

Moreover, STUMPF & KERLE (2011) summarized common textures used in landslide studies: entropy, contrast, correlation, mean and standard deviation (STUMPF & KERLE 2011:2568). Especially, the topographically-guided GLCM derivatives of contrast, correlation and entropy (Figure 2-13) showed good performance in their random forest algorithm (STUMPF & KERLE 2011:2571). However, in the frame of this study the following GLCM textures were investigated: entropy, contrast, inverse difference moment, and variance. Furthermore, they were derived by using the slope instead of using the DTM, which is more in common. Thus, in the following only the textures, which are used here, are briefly explained. More details can be found in HARALICK et al. (1973).

- *Entropy*: measurement for the loss of information or energy by a transmitted signal or physical transformation. Therefore, it shows the amount of information or energy of the image or object that is needed for the image or object compression (BLASCHKE et al. 2014a:4811, KUMAR & SREEKUMAR 2014:7670) (Figure 2-13).

- *Contrast*: measurement for the local variation present in the object or image (HARALICK et al. 1973:614).
- *Inverse difference moment (IDM)*: measures the local homogeneity of an object or image. High values represent uniformity in grey level and a high value of an inverse GLCM (KUMAR & SREEKUMAR 2014:7670).
- *Variance*: variance of the GLCM of an object or image (HARALICK et al. 1973:619).



Moving-window size: 5 m. Distance: 1 m. Anonymized extract (s. 3.2).

Figure 2-13: Texture entropy in flow direction (OWN REPRESENTATION).

## 2.1.5 Landslides

Landslides are natural events occurring on the whole world. When landslides cause great damage to human infrastructure and losses of human life, they become natural disasters (CRED definition, s. INTERNATIONAL FEDERATION OF RED CROSS AND RED CRESCENT SOCIETIES (2015:211)). According to the *World Disasters Report 2015*, in the period between 2005 to 2014 there were 170 reported landslide disasters causing 9,476 deaths, 1,978 million US\$ damage, and 3,173 thousands of affected people (INTERNATIONAL FEDERATION OF RED CROSS AND RED CRESCENT SOCIETIES 2015:220ff.).

In this chapter, initially the term landslide is defined. Subsequently, different types of landslides and triggers are briefly explained. A quick and easily understandable introduction of landslides is given by the landslide handbook of HIGHLAND & BOBROWSKY (2008). Detailed and fundamental information can be found in CRUDEN & VARNES (1996).

### 2.1.5.1 Definition

The definition of the term “landslide” slightly differs depending on the regarded discipline (HIGHLAND & BOBROWSKY 2008:4). VARNES (1978) published a landslide classification system which has become the most widely used system throughout the scientific world (HUNGR et al. 2014:167). Later, CRUDEN & VARNES (1996) reviewed the range of landslide processes and provided a vocabulary for describing different landslide features (CRUDEN & VARNES 1996:36). Recently, HUNGR et al. (2014) published an update of VARNES (1978) classification of landslide types by modifying the material aspect and related movement mechanisms (HUNGR et al. 2014:167).

For this thesis the landslide explanations of CRUDEN & VARNES (1996) are mainly used because they are still consistent with the standard terminology of the UNESCO Working Party (WPI/WLI 1993) (CRUDEN & VARNES 1996:37). Thus, the term landslide is defined according to CRUDEN & VARNES (1996) as “the movement of a mass of rock, debris or earth down a slope” (CRUDEN & VARNES 1996:36). Furthermore, they state that landslides are neither constrained to the land nor to sliding (CRUDEN & VARNES 1996:36).

### 2.1.5.2 Types

According to CRUDEN & VARNES (1996) there are different criteria for landslide identification and classification. The principal characteristics are, primarily, the type of movement, and, secondly, the type of material (CRUDEN & VARNES 1996:37). Additionally, other criteria such as rate of movement, state of activity, causes, features, geometry, or water content can be used for landslide description (VARNES 1978:11, CRUDEN & VARNES 1996).

The material of mass is divided in rock, debris, and earth and the five types of movement are: falls, topples, slides, spreads, and flows (CRUDEN & VARNES 1996:36). VARNES (1978) typology of complex landslides was dropped by CRUDEN & VARNES (1996), but the term “complex” was “retained as a description of the style of activity of landslide” (CRUDEN & VARNES 1996:37). In Figure 2-14 the different types of movement with respect to the material are paradigmatically illustrated.

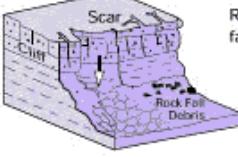
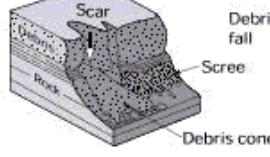
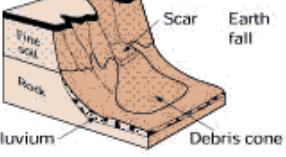
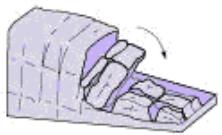
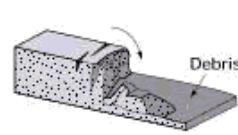
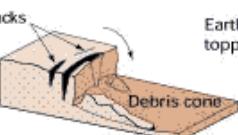
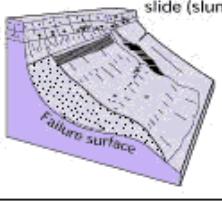
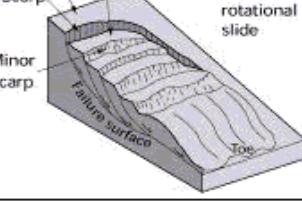
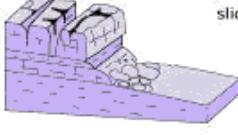
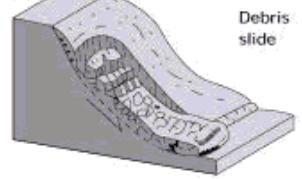
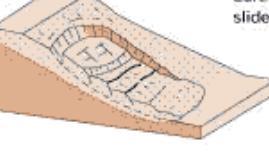
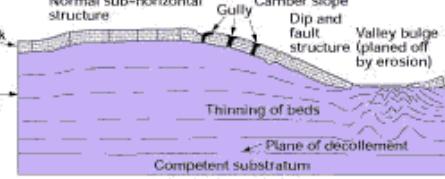
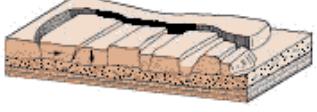
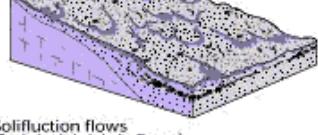
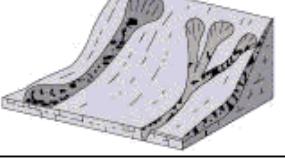
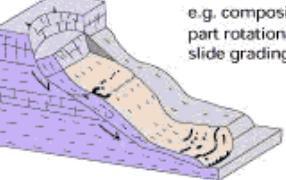
Material	ROCK	DEBRIS	EARTH
Movement type			
FALLS			
TOPPLES			
SLIDES	Rotational 	Crown Head Scarp Minor Scarp Failure surface Toe 	Successive rotational slides 
	Translational (Planar) 	Debris slide 	Earth slide 
SPREADS	Normal sub-horizontal structure Cap rock Clay shale Gully Camber slope Dip and fault structure Thinning of beds Plane of decollement Competent substratum 	e.g. cambering and valley bulging 	
FLOWS	Solifluction flows (Periglacial debris flows) 	Debris flow 	Earth flow (mud flow) 
COMPLEX	e.g. Slump-earthflow with rockfall debris. 	e.g. composite, non-circular part rotational/part translational slide grading to earthflow at toe 	

Figure 2-14: Landslide types by material and movement (BRITISH GEOLOGICAL SURVEY).

In the following the movement types are briefly explained:

- **Falls:** Downward movement through the air by falling, bouncing, or rolling of soil or rock from a steep slope surface with little or no shear displacement (CRUDEN & VARNES 1996:53).

- Topples: Forward rotational movement of a mass (soil or rock) out of the slope. Here, the axis or point of rotation is located below the center of gravity of the displaced mass (CRUDEN & VARNES 1996:54).
- Slides: Downward movement of a soil or rock mass along a slope surface of rupture or relative thin zones with intense shear displacement. CRUDEN & VARNES (1996) differentiate between two modes of slides: rotational and translational (CRUDEN & VARNES 1996:56).

The rotational slide is characterized by a surface of rupture with a concave curvature and a rotational movement on an axis being parallel to the contour of the slope. It is possible that the displaced mass can move along the rupture surface with little internal deformation. In addition, the head of the displaced material may occur with an almost vertically downward movement, whereas the upper surface may be backward-tilted towards the scarp (CRUDEN & VARNES 1996:56, HIGHLAND & BOBROWSKY 2008:10).

A translational slide mass moves out, or down and outward, along a planar or undulating surface of rupture. The displaced material may consist of loose, unconsolidated soils to extensive slabs of rock, or both. A translational slide is called block slide or planar slide when the mass, mostly rock, moves as single units (CRUDEN & VARNES 1996:57f., HIGHLAND & BOBROWSKY 2008:12).

In contrast to rotational slides, translational slides are relatively shallow and, if the surface of rupture is sufficiently inclined, they may continue moving. Instead, a rotational slide tries to find an equilibrium of the displaced mass (CRUDEN & VARNES 1996:57).

- Spreads: Extension of material - cohesive soil or rock mass, into a softer underlying material. The spreading takes place in combination with a general subsidence of the fractured mass. During the process there is no intense shear displacement on the rupture surface. Block, liquefaction, and lateral spreading can be distinguished as modes of spreading (CRUDEN & VARNES 1996:62, HIGHLAND & BOBROWSKY 2008:14).
- Flows: Movement of the displaced mass and inherent velocities are similar to a viscous liquid. Shear displacement on the rupture surface is of short lifetime, narrow and often not preserved. Depending on certain factors like water content, mobility, and evolution of the movement, an initial slide can become a flow (CRUDEN & VARNES 1996:64f., HIGHLAND & BOBROWSKY 2008:16).

### 2.1.5.3 Causes and Triggers

Landslides can occur under different geological, topographical, or climatic factors. They are seldom moncausal, and it is sometimes difficult to differentiate between landslide causes

and the potential final triggering mechanism. Therefore, the whole setting and context in which a landslide has taken place must be considered (VARNES 1978:26).

VARNES (1978) stated that "all slides involve the failure of the earth materials under shear stress" (VARNES 1978:26). He distinguishes between three major factors: 1. increase of shear stresses, 2. low shear strength, and 3. reduction of material strength (VARNES 1978:26ff.).

Such changes in shear strength originate from different causes. HIGHLAND & BOBROWSKY (2008) divided landslide causes into two categories: natural and human causes (HIGHLAND & BOBROWSKY 2008:30). Moreover, natural causes are sub-grouped into geological causes (susceptible or weathered materials), morphological causes (fluvial erosion of slope toe or tectonic uplift), and physical causes which often act as triggers (intense rainfall or rapid snowmelt) (CRUDEN & VARNES 1996:70, HIGHLAND & BOBROWSKY 2008:64). In Table 2-3 more causes and triggers are listed:

**Table 2-3: Further landslide causes and triggers (adapted from CRUDEN & VARNES 1996:70).**

human causes	geological causes	morphological causes	physical causes/triggers
▪ excavation of slope or its toe	▪ weak, sensitive or sheared materials	▪ volcanic uplift	▪ prolonged exceptional precipitation
▪ loading of slope or its crest	▪ jointed or fissured materials	▪ glacial rebound	▪ rapid drawdown (of flood and tides)
▪ drawdown (of reservoirs)	▪ adversely oriented mass or structural discontinuity (bedding, schistosity, fault, unconformity, contact, etc.)	▪ wave or glacial erosion of slope toe	▪ earthquakes
▪ deforestation		▪ erosion of lateral margins	▪ volcanic eruption
▪ irrigation		▪ subterranean erosion (solution, piping)	▪ thawing
▪ mining		▪ deposition loading slope or its crest	▪ freeze-and-thaw weathering
▪ artificial vibration		▪ vegetation removal (by forest fire, drought)	▪ shrink-and-swell weathering
▪ water leakage from utilities	▪ contrast in permeability or stiffness (stiff, dense material over plastic materials)		

## 2.2 State of the Art

In the last decade the availability of high-resolution digital terrain models by innovative progress in remote-sensing technologies lead to a significantly increasing number of applications and possibilities in landslide science (TAROLLI 2014:295). Especially, by the meanwhile lower cost availability of LiDAR DTMs derived from high quality LiDAR datasets and the progressed computer power, new approaches in landslide investigation emerged. JABOYEDOFF et al. (2012) distinguish these studies in four interrelated categories: 1. landslide detection and characterization, 2. hazard assessment and susceptibility mapping, 3. modelling, and 4. monitoring (JABOYEDOFF et al. 2012:10). Here, researches of the first and second category are of interest.

Landslide mapping was traditionally performed by visual interpretation of aerial photographs and/or intensive fieldwork. For large areas this task was, however, quite challenging, time-consuming and required expert knowledge that made the detection very subjective (MOINE et al. 2009:63). Since the 21st century this traditional mapping was increasingly supported and even replaced by the visual detection using LiDAR data. As one of the first relevant publications, SCHULZ (2004) and SCHULZ (2007) created a landslide inventory for Seattle by visually evaluating DTM derivatives including shaded relief, slope, and topographic contour maps. VAN DEN EECKHAUT et al. (2007) showed the potential of high-resolution data by mapping old deep-seated landslides even under forest area. More recently, PETSCHKO et al. (2016) detected landslides (earth and debris slides, earth flows, complex landslides, and areas with slides) by eye for statistical susceptibility modeling in Lower Austria. In these studies, for the manually landslide detection typical characteristics such as main and minor scarps, convex and concave slope areas, reverse slope, convex landslide foots, irregular hummocky topography, or alteration in drainage systems were considered (PETSCHKO et al. 2016:861, SCHULZ 2007:77, VAN DEN EECKHAUT et al. 2007:757).

MCKEAN & ROERING (2004) and GLENN et al. (2006) examined surface morphology of single landslides by using high-resolution LiDAR data to provide a more objective support for landslide mapping. In New Zealand near Christchurch MCKEAN & ROERING (2004) quantified local surface roughness of landslides by the local variability in aspect and slope using cosine eigenvalue ratios, by the mathematical Laplacian operator, and a two-dimensional spectral analysis (MCKEAN & ROERING 2004:337ff.). GLENN et al. (2006) differentiated morphological components (upper block, body, toe) within two canyon-rim landslides in southern Idaho by calculating surface roughness and slope semi-variances (GLENN et al. 2006:132f.).

Over the years, besides manually landslide detection, also semi or full automatic algorithms using high-resolution remote sensing imagery including LiDAR data emerged. In this context, mainly two different kind of approaches evolve: One the one hand mathematical and pixel-based, and on the other hand object-oriented and hierarchical-organized approaches.

BOOTH et al. (2009) developed an automated algorithm that correctly identified an average of 82% of the terrain of deep-seated landslides in the Tualatin Mountains, Oregon, and the Puget Sound lowlands, Washington. By applying two signal processing techniques - the 2D discrete Fourier transform and 2D the continuous wavelet transform, they could extract the typical landslide signature (scarp, hummocky topography, displaced blocks) out of the high-resolution topographic data (10 m resolution) (BOOTH et al. 2009:146). Moreover, the algorithm is available as free download in MATLAB format (BOOTH et al. 2009:137). The Wavelet transform approach was also adapted by KALBERMATTEN et al. (2012) and DOGLIONI & SIMEONE (2014) to analyze scale-dependencies of landscape features in a valley of the

Swiss Jura (KALBERMATTEN et al. 2012), and to detect large singularities in a medium resolution DEM as anomalous morphological structures for characterizing large deep-seated landslides in low Biferno valley, respectively (DOGLIONI & SIMEONE 2014). MANN et al. (2012) provided ideas and suggestions for a (semi) automatic extraction of shallow landslide features in a tropical forest using curvatures, a surface roughness index, and a vegetation index by combining ESRI Software ArcGIS tools and Python scripts. With the same software tools KAMP (2012) developed in her master thesis a semi-automatic landslide mapping toolbox using high-resolution LiDAR derivatives in a study region of Styria, Austria.

However, with the availability of high-resolution data pixel-based approaches are often confronted with noise inherent in the data. Geographic object-based image analysis (GEOBIA) tries to overcome this challenge. Here, the image is segmented in regions with same characteristics in the first step (s. 4.1.1), approaching the way humans perceive their environment. The most landslide studies combined high-resolution optical, multispectral data and DTM derivatives for their analyses: MARTHA et al. (2010) used spectral, spatial and morphometric properties of landslides and knowledge-based thresholds to detect five landslide types (debris slides, debris flows, shallow translational/rotational/translational rock slides) in an area in the Himalayas, India. Their method was semi-automatically and achieved accuracies of 76.4% in landslide recognition and 69.1% in landslide classification. MARTHA et al. (2011) extended the approach of MARTHA et al. (2010) by using K-means thresholding for classification and a multiscale segmentation with optimal parameters derived from an objective function (s. 4.1.1.4). In such a way, they could improve the overall recognition accuracy to 76.9% and the classification accuracy to 77.7%. LU et al. (2011) developed a rapid semi-automatic mapping technique for new landslides after major events. On the Messina landslide event in southern Italy on the first of October 2009 they could achieve an 81.8% user accuracy for the number of landslides, and 75.9% for their extent by using temporal principal component analysis, image matching through spectral angle mapper, anomaly detection by the Reed-Xiaoli detector, and textural analysis with gray-level co-occurrence matrix for their landslide detection approach. For four sites in the northern hemisphere (Haiti, Italy, China, France) STUMPF & KERLE (2011) tested successfully the machine learning algorithm Random Forest with a set of more than 77 object metrics (accuracies between 73% and 87%). More recently, BLASCHKE et al. (2014a) examined the use of topographically oriented GLCMs in combination with spectral and shape-based information to derive landslide areas in the Urmia Lake basin in northwestern Iran. However, the above-mentioned object-oriented approaches have the limitation that they are depended on the availability of optical images, and can therefore not detect landslides under forest area. Moreover, they classified individual landslides directly in its entirety, instead of different landslide parts. VAN DEN EECKHAUT et al. (2012) investigated the object-oriented detection of deep-seated landslides by only using a DTM and its deriva-

tives in the Flemish Ardennes, Belgium. The semi-automatic approach detected different landslide parts - main scarp, flanks, and body by their specific morphometric characteristic with supervised support vector machines at an appropriate scale. The different landslide parts grew ultimately together into a single landslide. As a result, approximately 70% of the landslide body and 90% of the main scarp could correctly be detected.

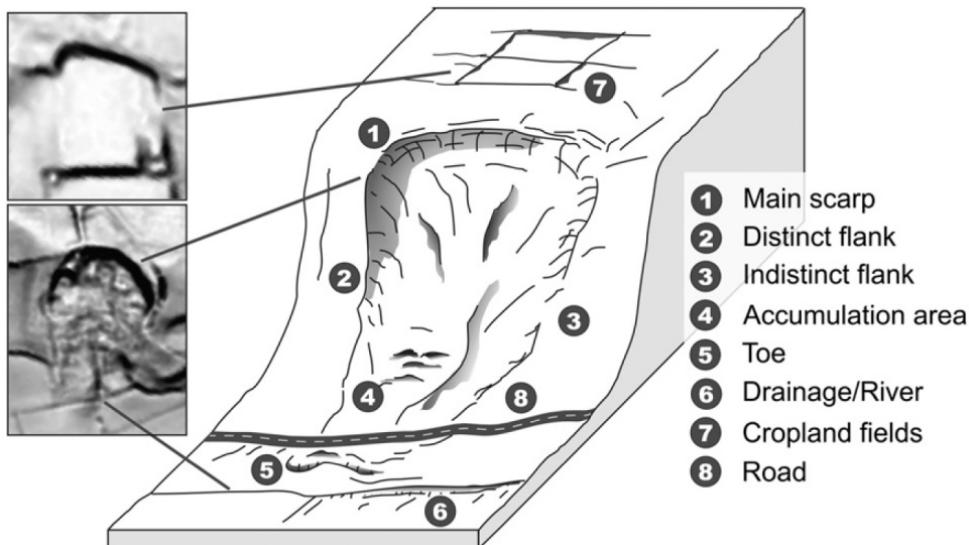
For the object-oriented approaches the authors mentioned-above used mainly the commercial software Definiens eCognition to develop rulesets for landslide detection.

### 2.3 Research Question

Build on the work of MARTHA et al. (2011) and VAN DEN EECKHAUT et al. (2012) the objective of this Master thesis was the development of an object-oriented approach that automatically detects landslides by using only high-resolution LiDAR DTM data derivatives in a study area within the federal state of Burgenland, Austria. To avoid expensive commercial software such as Definiens eCognition or ESRI ArcGIS, only open-source products were investigated according to their functionality and capability with regard to the research objective. Moreover, only such software had been of interest, that could be integrated into the free development environment R.

By combination of passive optical images with high-resolution LiDAR derivatives, landslides were generally segmented as one or few segments (c.f. MARTHA et al. 2011, STUMPF & KERLE 2011). However, VAN DEN EECKHAUT et al. (2012) stated that this is not the case by considering only morphometric information (Figure 2-15). Landslides are too geomorphologically complex to represent them by one single segment. Instead, they consist of different features that can be identified by their appropriate scale, and subsequently aggregate into one single landslide segment (VAN DEN EECKHAUT et al. 2012:40).

In this thesis, the approach of VAN DEN EECKHAUT et al. (2012) was revisited and modified in a way that for the classification k-mean thresholding (c.f. MARTHA et al. 2011) instead of supervised vector machine was used. In addition, besides the possibilities of land surface parameters, also shape-based information and topographically-oriented textures were investigated for this task.



**Figure 2-15: Schematic composition of a landslide (van den EECKHAUT et al. 2012:32).**

Therefore, the objectives of this thesis can be summarized as follows:

- automated detection of scarps, flanks and landslide bodies (Figure 2-15) (s. 4.1.2),
- object-oriented analysis (s. 4.1.1),
- investigation of optimal and objective segmentation scale parameters (s. 4.1.1.4),
- investigation of automatically derived thresholds for the separability of possible false positive candidates,
- investigation of suitable land surface parameters, shape-based information and textures,
- estimation of detection rate (s. 4.1.4 and 5.2),
- use of open-source products only (s. 4.2).

In this thesis only the CRUDEN & VARNES landslide types of slides and flows (s. 2.1.5.2, Figure 2-14) were investigated. Therefore, continuing from here, these types are addressed when using the more general term landslide.

### 3 Study Area and Related Data

#### 3.1 Study Area in Burgenland

The study area is located in Austria most eastern province, the Federal State of Burgenland. As described in LEOPOLD (2012) and LEOPOLD et al. (2013) most of the province's area is dominated by basin sediments of Neogen and Quaternary age mainly consisting of clay, silt, sand and gravel. Additionally, the clay material is often over-consolidated. This geological composition makes the area very prone for landslides (LEOPOLD et al. 2013:75).

According to LEOPOLD et al. (2015) the federal state government of Burgenland had increasingly registered damages on buildings and infrastructure due to landslides in the last decade. These can be attributed to more frequently heavy precipitation events due to regional climate change, landslide-prone slopes due to existing geology, and the sprawl of settlement into landslide-prone areas (LEOPOLD et al. 2015:5).

Due to the availability of high-resolution LiDAR data and the high variability of geological units, an area in the district Oberpullendorf was selected as study area.

The study area had a rectangular extent from 16°18'37,825"E 47°28'21,052"N to 16°30'4,449"E 47°24'50,124"N with the MGI/Austria GK East as spatial reference (EPSG: 31256), and, hence, encompassed around 93.54 km<sup>2</sup> (Figure 3-1). According to the CORINE Land Cover (CLC) dataset from the year 2012 around 68.53 % of the study area comprised forest and semi natural areas, less than one third was under agricultural use, and only 2.87 % were artificial surfaces, namely urban fabric (Table 3-1).

**Table 3-1: Summarized land cover of the study area (OWN REPRESENTATION).**

CLC code	label	area (km <sup>2</sup> )	area (%)
112	artificial surfaces	2.682	2.87
211, 231, 242, 243	agricultural areas	26.752	28.60
311, 312, 313, 324	forest and semi natural areas	64.101	68.53

As land cover classification the CORINE Land Cover 2012 was used.

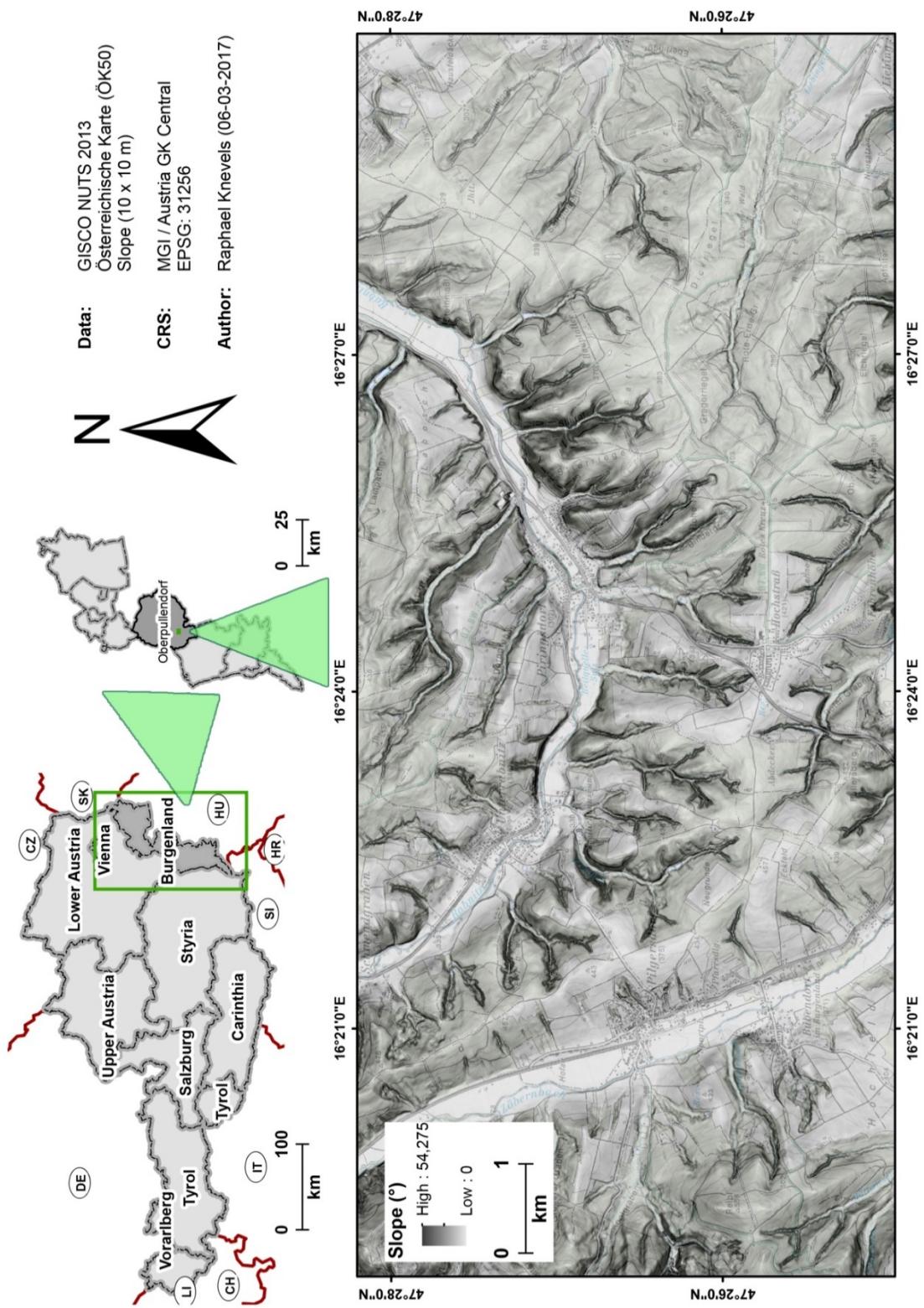


Figure 3-1: Study Area – Overview (OWN REPRESENTATION).

### 3.2 Digital Terrain Model

The LiDAR data was commissioned by the Federal State Government of Burgenland, and the digital terrain model of the district of Oberpullendorf for this thesis was provided by the AIT Austrian Institute of Technology GmbH.

According to the e-mail correspondence with ZALKA (2016) the study area was scanned between the 23-11-2010 and 10-04-2011 during the leaf-off period by overflights in 400 to 500 m height above ground. Therefore, the RIEGL laser scanner measurements system of LMS-Q560 (Hk36) and LMS-Q650 (DA42) were used. The average airspeed was around 148 km/h with a mean scan speed of 118 lines/s for the Hk35 overflight, and 204 km/h and 118 lines/s for the DA42 overflight, respectively. In average more than 10 points per m<sup>2</sup> were recorded as last return during the flights (ZALKA 2016). More technical details about the used airborne laser scanners are available in the data sheet of RIEGL (2010) and RIEGL (2012).

Furthermore, ZALKA (2016) stated that the DTM was derived by the vendor with the Terrasolid software TerraModeler using the ground points of the LiDAR point cloud. The resulted grid resolution is 1 x 1 m. There has not been an accuracy assessment for the DTM conducted, yet. However, the vertical and horizontal accuracy of the LiDAR point cloud in comparison to planes of reference measured in the residues was less than 6 cm for three standard deviations (ZALKA 2016).

In the study area the elevation of the DTM ranged from 248.97 m to 607.24 m with an average height of 376.55 m. An area of around 10,743 m<sup>2</sup> were signed as no data.

Due to restriction clauses, in this thesis it is not allowed to publish any high-resolution LiDAR data and its derivatives such as slope or landslide products as maps with absolute position information. Instead, anonymized extracts are illustrated or resampled data to 10 x 10 m resolution is shown (Figure 3-1).

### 3.3 Landslide Inventory

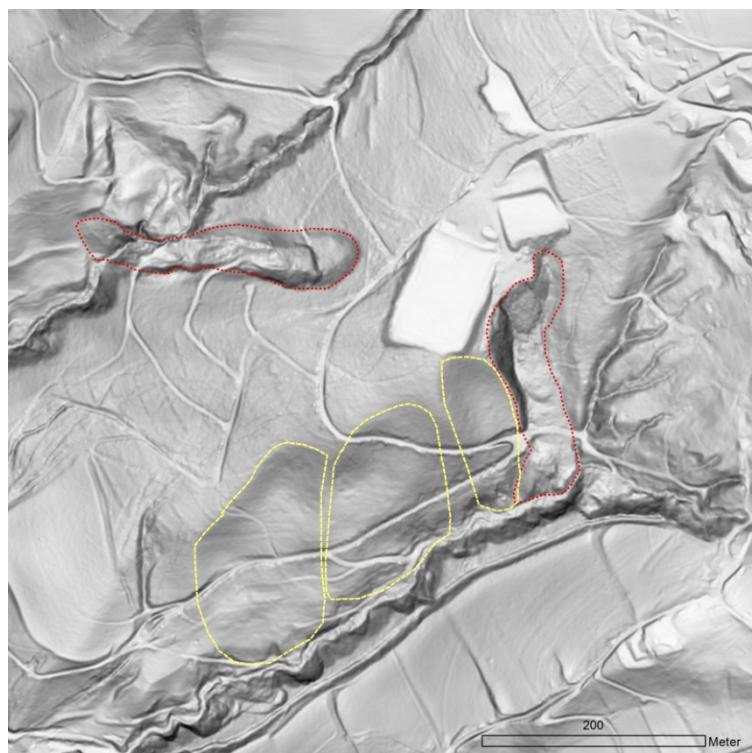
According to GUZZETTI et al. (2012) landslide inventories can be grouped into archive and geomorphological inventories. While the former consists of landslide information obtained from the literature, or other archive sources, the latter can be further classified in historical, event, seasonal or multi-temporal inventories (GUZZETTI et al. 2012:45). The landslide inventory resulting in LEOPOLD et al. (2015) in the district of Oberpullendorf is an historical inventory. A historical inventory is defined as an inventory containing landslide events over time in a region with no or relative differentiation of the landslide age (GUZZETTI et al. 2012:45, MALAMUD et al. 2004:687).

For the first time, landslides were investigated in the district of Oberpullendorf in 2007 and published in LEOPOLD (2012). Here, 21 active landslides could be detected by intensive fieldwork (LEOPOLD et al. 2015:5, 2015:16). In 2014 the landslide investigation was revised based on the currently available high-resolution DTM (s. 3.2). With this revision, the inventory could be extended by additional 428 landslide events (LEOPOLD et al. 2015:16). As landslides only the CRUDEN & VARNES (1996) types flows and slides were investigated (s. 2.1.5.2), but were not precisely attributed in the inventory database.

The visually mapping method was performed according to SCHULZ (2004) using slope and hill-shaded maps as well as orthophotos. A landslide was mapped when the following conditions were fulfilled (LEOPOLD et al. 2015:16f.):

- All morphological structures of a landslide (scarp, body) must be clearly recognizable.
- If only the scarp but no body with moved material was visible, then the landslide was not recorded.
- Only landslides with relatively young morphological structures were finally kept in the inventory. To differentiate ages of morphological structures of landslides the schematic figure of BELL et al. (2012) was added for the mapping (s. BELL et al. 2012:138, Figure 3-2).

Afterwards the landslide inventory was – if necessary – corrected by random site inspections (LEOPOLD et al. 2015:16f.).



**Red: landslides with young structures. Yellow: landslides with old structures (not recorded in inventory)**  
**Figure 3-2: Landslides with different ages in their morphological structure (LEOPOLD et al. 2015:18).**

The study area comprises 148 of the 428 landslides events in the district of Oberpullendorf. The landslide area ranged from 0.04 ha to 17.22 ha (mean: 1.23 ha) with a mean slope angle of 18.34°. In approximately 90.25% of the landslides were mapped on forest and semi natural areas, while around 5.97% are located on agricultural areas and 3.78% on areas labeled as artificial surfaces in the CLC2012 classification. Further statistics are shown in Table 3-2.

Until now, there has been no quality assessment of the inventory taken place. However, LEOPOLD (2016) estimated that approximately 95% of the visible landslides in the DTM of the study area were mapped.

**Table 3-2: Landslide inventory statistics for the study area (own representation).**

parameter	min	max	mean	sum	agricultural areas	forest and semi natural areas	artificial surfaces
area (ha)	0.04	17.22	1.23	178.86	10.68	161.42	6.76
perimeter (m)	77.01	1820.58	397.73				
slope (mean, °)	7.80	29.46	18.34				

## 4 Methodology

### 4.1 Theory of Automatic Landslide Detection

#### 4.1.1 Object-Oriented Image Analysis

In this chapter, the fundamentals of object-oriented image analysis are briefly explained and linked to the open source software used in this thesis. As basic literature BLASCHKE et al. (2008)'s "Object-based image analysis - Spatial concepts for knowledge-driven remote sensing applications" is to be mentioned.

##### 4.1.1.1 Introduction

In the last 15 to 20 years object-oriented or object-based image analysis (OBIA) became very popular in remote sensing, however, the approach has its roots in industrial image processing already in the 1980s (BLASCHKE 2010:3). The principal idea of OBIA is to segment, cluster or regionalize the image into distinguishable "meaningful objects" by their homogeneous characteristics. The objective is the reduction of the level of detail and complexity inherent in the image and the creation of object attributes such as area, shape, statistical parameters, texture or spatial relationships between these objects for further analysis (LANG 2008:6, BLASCHKE et al. 2014b:180ff.). An image object can therefore represent a real-world object – emulating the way humans interpret images, which a pixel cannot (HAY & CASTILLA 2008:76, LANG 2008:13). OBIA is particularly meaningful with regard to high resolution data when the pixel size of an image is significantly smaller than the average size of an object of interest (LANG 2008:6).

OBIA is also performed outside the remote sensing community, namely in industrial, biological, or medical image processing (BLASCHKE 2010:3). To distinguish from other disciplines HAY & CASTILLA (2008) proposed and defined the term Geographic Object-Based Image Analysis (GEOBIA) as a sub-discipline of Geographic Information Science (GIScience) (HAY & CASTILLA 2008:77). Moreover, HAY & CASTILLA (2008) claimed GEOBIA as a new paradigm shift in remote sensing image analysis, which is further discussed and refined in BLASCHKE et al. (2014b) (HAY & CASTILLA 2008:75).

BAATZ et al. (2008) proposed to distinguish between the terms object-based and object-oriented image analysis. Former means that the created objects will not be further modified and serve as basis for further analysis (BAATZ et al. 2008:30). In the latter instead, objects serve "not only as information carriers but also as building blocks for any further shape modification, merging, or segmentation procedures" (BAATZ et al. 2008:31). However, according to BLASCHKE (2010), in literature the terms are found to be used simultaneously with the tendency to the term "based" to avoid the close link to object-oriented programming paradigm (BLASCHKE 2010:4).

The linkage between objects and land surface parameters derived from a digital terrain model on different scales was already proposed by DIKAU (1989, 1990). DIKAU (1990) supported the idea of a geomorphological object being complex and consisting of multi-scale systems that can be analyzed by methods of hierarchical theory (DIKAU 1990:232). The object-oriented approach enables the possibility to detect different landslide parts on multiple-scales by using different land surface parameters, and therefore this approach was used in this thesis. Because of the availability of hierarchical segmentation algorithms, the software GRASS GIS was used to perform multi-scale analysis.

#### 4.1.1.2 Segmentation Algorithms

According to BLASCHKE et al. (2014b) image segmentation is defined as “the partitioning of an array of measurements on the basis of homogeneity [...] into spatially continuous, disjoint and homogeneous regions referred to as ‘segments’” (BLASCHKE et al. 2014b:186). There are many algorithms for segmenting an image. An introduction to this topic is given by BLASCHKE et al. (2004:219ff.).

Three commonly distinguished traditional image segmentation approaches exist (BLASCHKE et al. 2004:217): pixel-, edge- and region-based segmentation methods.

- Pixel-based methods use image thresholding and segmentation in feature space, sometimes referred to as property space.
- In edge-based segmentation, commonly, as first step edges are found by edge detection, and are subsequently used to delineate boundaries between image objects.
- Region-based approaches can be furthermore subdivided into region growing, region merging and splitting methods, and their combinations.

Many region growing algorithms start growing of the region by a set of seed points. Neighborhood pixels of these seeds are subsequently added, or not, to the region of the seed depending on given homogeneity criteria. The algorithm terminates when all pixels of the image are assigned to a specific seed region.

The region merging and splitting method splits or merges image objects depending on the homogeneity criteria. The splitting part divides the image or image objects in smaller sub-regions if the homogeneity criteria are not fulfilled. The merging part groups smaller segments into larger segments to fulfill the homogeneity criteria (BLASCHKE et al. 2004:217f.).

In SAGA and GRASS GIS a variation of the region growing algorithm is implemented.

GRASS GIS uses the region growing approach of SPRING (Sistema de Processamento de Informações Georreferenciadas) which has been developed at the National Institute for

Space Research (INPE). The algorithm can be accessed through the function *i.segment*. Important parameters controlling the segmentation result are: 1. *threshold*, which determines the allowed homogeneity distinction for merging a neighbor pixel to an object, 2. *minsize*, which fixes the minimum size an object can have, and 3. *seeds* (optional), which provide starting segments for the algorithm. Moreover, there is the possibility to add a boundary grid to limit the segmentation for a region or class. By the use of seeds a bottom-up and by the use of bounds a top-down multiscale or hierarchical segmentation is enabled. Further details of the algorithm can be found in BINS et al. (1996).

In SAGA GIS a seeded region growing algorithm is implemented and described in details in ADAMS & BISCHOF (1994) and BECHTEL et al. (2008). As the name of the algorithm already indicates, it depends on a seed input to start the segmentation. The quality of the seeds thus determines the result of the segmentation. Besides, other parameters controlling the segmentation result are: 1. *method*, which is distinguished in variance in feature space and position, or in variance in feature space, and 2. *similarity threshold*, which set the allowed homogeneity distinction for merging a pixel to an object.

In the last decade, besides the traditional image segmentation approaches multiple so-called superpixel algorithms emerged for image segmentation. A superpixel algorithm divides an image into a given number of segments that are called superpixels. According to ACHANTA et al. (2012) there are two kind of algorithms for generating superpixels: graph-based or gradient ascent methods (ACHANTA et al. 2012:2274). As GRASS GIS add-on the simple linear iterative clustering (SLIC) algorithm can be accessed by the function *i.superpixels.slic*. SLIC belongs to the category of gradient-ascent based algorithms. Initially, cluster centers are sampled as pixels at the regular grid. Then, the cluster is iteratively refined by gradient ascent methods until some truncation condition is reached to form a superpixel (ACHANTA et al. 2012:2275). Important parameters controlling the segmentation result are: 1. *num\_pixels*, which defines the initial number of cluster centers for superpixels, and 2. *compactness*, which determines the compactness as truncation condition for clustering pixels to an object. The SLIC algorithm is explained in full detailed in ACHANTA et al. (2012).

In the frame of this thesis the SAGA and GRASS GIS as well as the SLIC segmentation algorithms were implemented in the *Segmentation* function (s. 4.3.1).

#### 4.1.1.3 Derivation of Seeds

Dependent on the algorithm, seeds can be single pixel points, cluster, or even a segmented image. They serve as starting points for growing or clustering techniques that merge neighboring pixels to objects. BÖHNER et al. (2006) denote the computation of suitable seeds as

crucial task, since the quality of the seeds also affect the quality of the segmentation (BÖHNER et al. 2006:34).

In SAGA GIS the seeds can be computed by the modules *Seed Generation* or *Fast Representativeness*. In *Seed Generation* seed pixels are chosen and set as minimum or maximum of variance in a user-defined size of either a local window, or a local search function with weighting. In the module *Fast Representativeness* seed pixels are determined by a geostatistical approach: a fitted semi-variogram function is used to calculate the spatial representativeness of a pixel feature (BÖHNER et al. 2006:31f.). The amount of objects resulting by the segmentation is affected by the option of *Level of Generalisation*.

Related to the recent version used in this thesis, GRASS GIS does not offer a module for computing seeds. However, in *i.segment* there is the possibility to use a seed grid as input. The seed grid can be a resulted segmentation of a previous *i.segment* procedure, the SLIC or SAGA seeded region growing algorithm (s. 4.3.1).

In the GRASS GIS add-on *i.superpixels.slic* the initial cluster centers (or seeds), that define the superpixels can be refined by the parameter *perturb*. According to ACHANTA et al. (2012) the option allows the user to set a local window around the initial cluster centers. Subsequently, the seeds are moved to the location corresponding to the lowest gradient. By this, the centering of a superpixel in a nosy environment or on an edge, is avoided (ACHANTA et al. 2012:2275).

#### 4.1.1.4 Optimization of Scale Parameters

For object-oriented image analysis the segmentation is a crucial pre-processing step. There are different scale parameters such as similarity threshold, min-size, or compactness controlling the output of the segmentation. The selection of the appropriate scale parameter is challenging and often underlying more subjective trial-and-error methods (DRĂGUȚ et al. 2010:860). However, there is an active discussion about how to assess image segmentation quality and objectively optimize segmentation scale parameters (DRĂGUȚ et al. 2014, NEUBERT et al. 2008).

There are basically two approaches implemented in common GIS software using so-called unsupervised segmentation parameter optimization (USPO): the estimate scale parameter (ESP) tool of DRĂGUȚ et al. (2010) which can be used as add-on in Definiens eCognition, and the objective function of ESPINDOLA et al. (2006) which can be accessed through the GRASS GIS add-on *i.segment.uspo*. In the following, the fundamentals of the approaches are briefly explained.

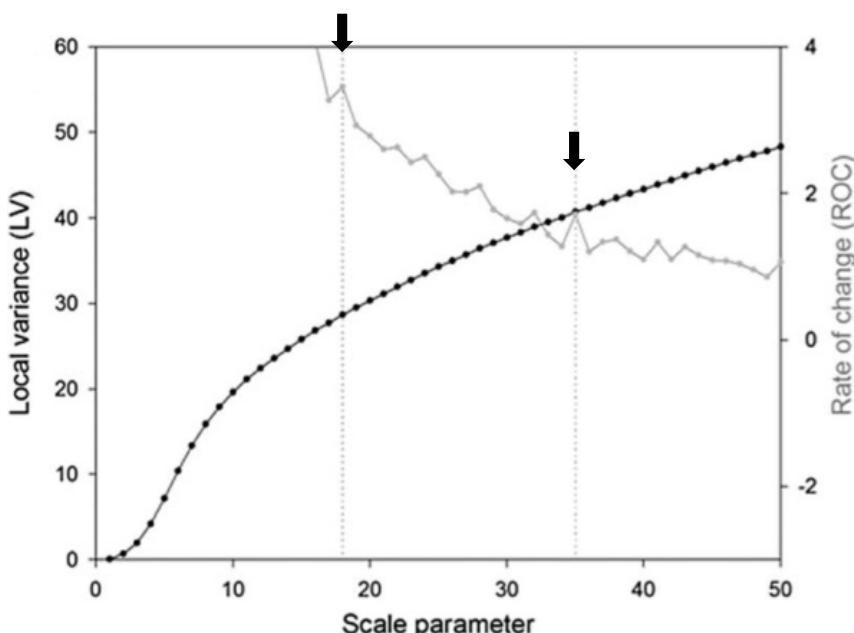
*ESP*. According to DRĂGUȚ et al. (2010) the ESP tool is based on the idea of the relationship between the spatial structure, the size of a real-world object, and the image resolution

(DRĂGUȚ et al. 2010:860). For every segmentation scale the local variance (LV) as mean standard deviation of the objects is computed. Starting from small-scale parameter the LV increase with growing segment sizes. If the segment size fits well the real-world objects, DRĂGUȚ et al. (2010) assume that the segment boundaries will be conserved even on a certain number of further coarser segmentation scales (DRĂGUȚ et al. 2010:861).

The change of LV between the different segmentation scales is measured by the rate of change of local variance (ROC-LV) function (DRĂGUȚ et al. 2010:863):

$$ROC\ LV = \left[ \frac{L - (L - 1)}{L - 1} \right] * 100$$

With  $L$  representing LV at target segmentation scale, and  $L - 1$  representing LV at next lower segmentation scale. By plotting ROC-LV and LV against segmentation scale on a graph, break points or peaks indicate optimal scale parameters creating “meaningful” objects (Figure 4-1).



Gray line: ROC-LV. Black line: local variance. Dotted vertical lines and arrows: optimal scale parameters.

Figure 4-1: ESP-Tool tool output example for temporary human settlements (DRĂGUȚ et al. 2010:864).

The approach of DRĂGUȚ et al. (2010) was further refined by DRĂGUȚ et al. (2014) to automatically extract optimal scale parameters out of the ROC-LV function.

*Objective Function.* The objective function of ESPINDOLA et al. (2006) is based on the idea that a segmentation has two desirable properties: the objects of the segmentation results should be internally homogeneous and distinguishable from its surrounding objects (ESPINDOLA et al. 2006:3025). For the measurement of the intrasegment homogeneity a variance indicator and of the intersegment heterogeneity a spatial autocorrelation index are computed and combined.

The intrasegment homogeneity is calculated as weighted mean variance ( $\nu$ ) of all objects of a segmentation step by ESPINDOLA et al. (2006:3037):

$$\nu = \frac{\sum_{i=1}^n \alpha_i * \nu_i}{\sum_{i=1}^n \alpha_i}$$

With  $\nu_i$  as variance and  $\alpha_i$  as area of object  $i$ . By using the object sizes as weights, small objects have less potential to cause bias in the calculation. As a result, small variance values are desirable indicating large internally homogeneity of the objects.

For the computation of the intersegment heterogeneity the Moran's I ( $I$ ) is used as autocorrelation index, and can be expressed by following formula (ESPINDOLA et al. 2006:3037):

$$I = \frac{n \sum_{i=1}^n \sum_{j=1}^n w_{ij} (y_i - \bar{y})(y_j - \bar{y})}{(\sum_{i=1}^n (y_i - \bar{y})(\sum_{i \neq j} \sum w_{ij})}$$

With  $n$  as total number of objects,  $w_{ij}$  as measure for spatial proximity between the objects  $i$  and  $j$ ,  $y_i$  as mean value of object  $i$ , and  $\bar{y}$  as mean value of the segmentation step. As a result, small Moran's I values are desirable indicating small autocorrelation, and thus objects with clear boundaries that statistically differ from its neighbors (ESPINDOLA et al. 2006:3038).

To combine both statistical parameters they are first normalized in the sense that the smallest value becomes 1 and the largest value becomes 0. This can be expressed by the following formula (ESPINDOLA et al. 2006:3038):

$$F(x) = \frac{X_{max} - X}{X_{max} - X_{min}}$$

Finally, the objective function is the resulted sum of both normalized parameter values (ESPINDOLA et al. 2006:3038):

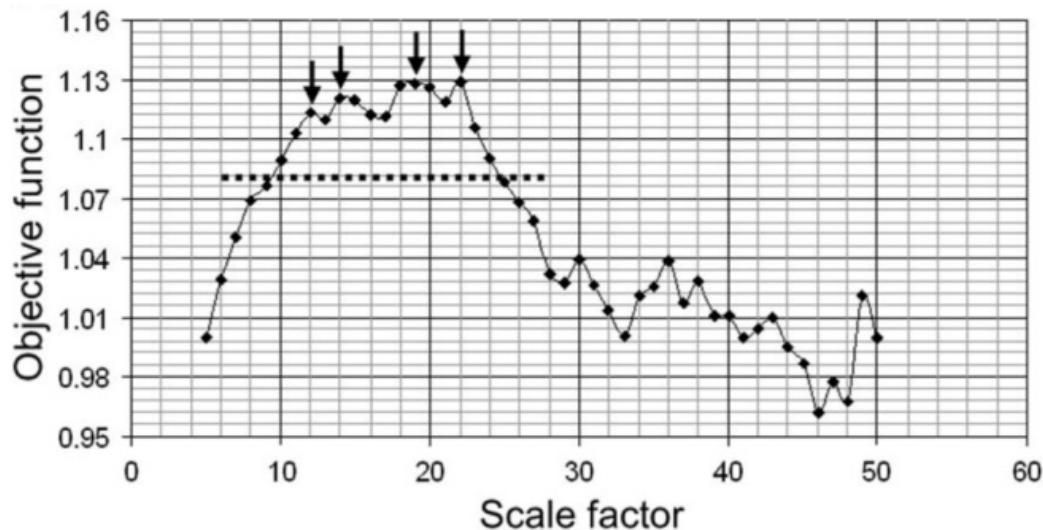
$$F(\nu, I) = F(\nu) + F(i)$$

The maximum value of  $F(\nu, I)$  gives information on the optimal segmentation scale parameters for the tested scales. However, MARTHA et al. (2011) demur that a single optimal scale is not capable to reflect the complexity between the spatial structure of an image and of a landscape (MARTHA et al. 2011:4931f.). Therefore, they propose to add a plateau objective function to receive multiple optimal segmentation scale parameters (Figure 4-2) (MARTHA et al. 2011:4932):

$$F(plateau) = F(\nu, i)_{max} - \sigma$$

With  $F(\nu, i)_{max}$  as maximum value and  $\sigma$  as standard deviation of the objective function for all tested scales. Scale parameters above the plateau objective function are regarded as close enough to the maximum value of the objective function to keep the balance between under- and over-segmentation. Furthermore, peaks in the graph (black arrows in Figure 4-2)

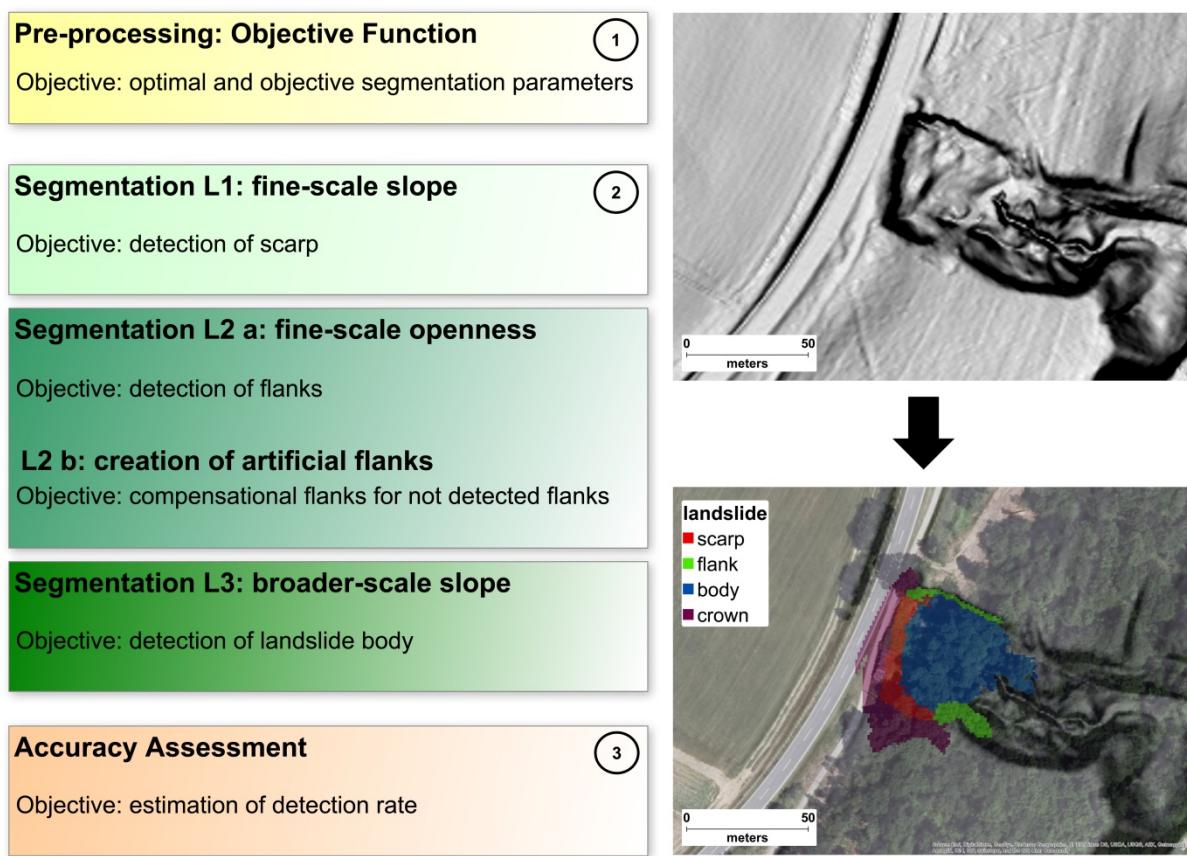
show locally optimal scale parameters in respect to their neighbors (MARTHA et al. 2011:4932).



Dashed black line: plateau objective function. Black arrows: peaks for locally optimal scale parameters.  
Figure 4-2: Objective function output example for Darjeeling (MARTHA et al. 2011:4934).

In the frame of this thesis both approaches were implemented in the R *Lslide* package. However, the objective function was finally chosen to estimate optimal scale parameters, and therefore its implementation is further explained in chapter 4.3.2. The choice was made for several reasons: 1. By the use of spatial autocorrelation and variance the objective function combines two robust and well-established statistical parameters. 2. SAGA and GRASS GIS segmentations can result in very small objects even consisting of one single pixel. By weighting the variance with the object sizes, the bias resulting of small objects is avoided. And 3. MARTHA et al. (2011) and VAN DEN EECKHAUT et al. (2012) also used the objective function in their studies, and could show their potential to estimate meaningful scale parameters for landslide detection in high-resolution data (s. 2.2).

#### 4.1.2 Detection of Landslides: Idea and Workflow



1: pre-processing. 2: three-level segmentation. 3: accuracy assessment.  
Orthophoto from ESRI's basemap (World Imagery).

Figure 4-3: Concept of landslide detection (OWN REPRESENTATION).

In the high-resolution DTM the different landslide parts are clearly distinguishable by their geomorphometric characteristics. By visual interpretation of suitable images, we can recognize landslide parts by their shape, texture, embedding in their environment and their value (i.e. slope). The idea was to translate the visual recognition characteristics into a suitable ruleset that could be computed by available open source software.

Due to different characteristics of the landslide, it was not possible to simply detect the landslide as “a whole” in one single segmentation level. Instead, the different landslide parts must be investigated in multiple levels by hierarchical segmentation computed with suitable scale parameters.

Therefore, the workflow of this thesis was divided into three sections: 1. pre-processing, 2. hierarchical segmentation, and 3. accuracy assessment (Figure 4-3).

In the first part, multiple segmentation scale parameters were investigated by the objective function. The function provides objectively estimated, optimal scale parameters that could be further used in the segmentation hierarchy of the second part.

The second and main part was subdivided into three interconnected segmentation levels. Each level had the objective to detect a different landslide part by using different segmentation scales and input parameters. Segmentation level 1 (L1) had the objective to detect potential landslide scarp candidates. In level 2 (L2) flanks and crowns should be detected. The landslide body was investigated in level 3 (L3). In addition, on this level an attempt was made to finally clean false-positive candidates. A landslide was found when at least a scarp and a belonging landslide body were detected.

To receive information on the quality of the landslide detection, the accuracy of the classification was assessed in the third part.

To detect a landslide, the morphometric characterization of each part must be clear, and suitable land surface parameters must be chosen. Table 4-1 is adapted from VAN DEN EECKHAUT et al. (2012) and shows the typical individual characteristics as well as their possible false positive candidates (s. Figure 2-15).

**Table 4-1: morphometric characterization of different landslide parts (adapted from VAN DEN EECKHAUT et al. (2012:33)).**

landslide part	characteristics	possible false positive	characteristics
<b>scarp</b>	<ul style="list-style-type: none"> <li>▪ steep</li> <li>▪ semi-circular</li> <li>▪ concave in planform</li> <li>▪ main direction <math>\perp</math> flow direction</li> <li>▪ area downslope has high surface roughness</li> </ul>	<i>Earth banks (roads and cropland fields)</i>	<ul style="list-style-type: none"> <li>▪ steep</li> <li>▪ area downslope has often low surface roughness</li> </ul>
<b>flanks</b>	<ul style="list-style-type: none"> <li>▪ main direction <math>\perp</math> main direction of scarp</li> <li>▪ upslope part has significantly higher elevation than landslide body</li> <li>▪ downslope part is less clear</li> </ul>		
<b>landslide body</b>	<ul style="list-style-type: none"> <li>▪ high surface roughness</li> <li>▪ downslope of scarp bordered by flanks</li> <li>▪ debris is displaced in downslope direction</li> </ul>	<i>anthropogenically altered slope sections</i>	<ul style="list-style-type: none"> <li>▪ high surface roughness</li> <li>▪ various size and shape</li> </ul>
<b>foot</b>	<ul style="list-style-type: none"> <li>▪ might border a river</li> <li>▪ generally, not clear</li> </ul>		

In Table 4-2 the most important parameter used for the detection are shown. All parameters are available in open source software and could be addressed by corresponding packages in R. Parameters that are not a part of packages such as the sky view factor or  $D^\infty$  flow could be computed by *system()*-commands in R.

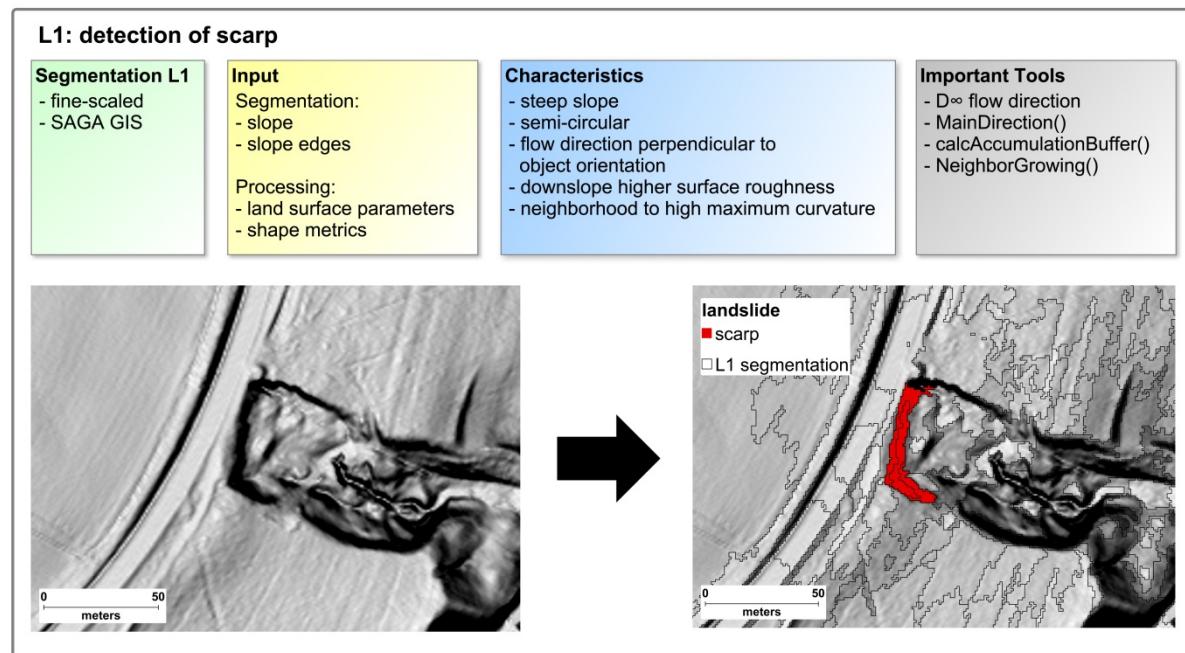
**Table 4-2: Overview of parameters used for landslide detection (own representation).**

parameter	method	settings/formula	software	module
<b>land surface parameters</b>				
normalized height	DIETRICH & BÖHNER (2008)	w = 5; t = 2; e = 2 (s. 2.1.2.2)	SAGA GIS	Relative Heights and Slope Positions
slope	ZEVENBERGEN & THORNE (1987)		SAGA GIS	Slope, Aspect, Curvature
slope edges			SAGA GIS	Edge Detection (ViGrA)
maximum curvature	WOOD (1996)	WS 31	SAGA GIS	Morphometric Features
D $\infty$ flow	TARBOTON (1997)		TauDEM	DinfFlowDir
openness	VAN DEN EECKHAUT et al. (2012)	WS 51	GRASS GIS	r.neighbors
sky-view factor	ZAKŠEK et al. (2011)	SR 20; D 16	Relief Visualization Toolbox	skyViewFactor (s. 4.3.7)
surface roughness	SAPPINGTON et al. (2007)	WS 15	GRASS GIS (add-on)	r.vector.ruggedness
<b>textural features</b>				
entropy, variance, ...	HARALICK et al. (1973), STUMPF & KERLE (2011)	WS 5; Dst 1	GRASS GIS	r.texture, TextureFlow (s. 4.3.6)
<b>shape metrics</b>				
shape indices	LANG & BLASCHKE (2007)		SAGA GIS	Polygon Shape Indices
convex hull			SAGA GIS	Convex Hull
compactness	VAN DER WERFF & VAN DER MEER (2008)	$4 \pi \frac{\text{Area}(S)}{\text{Perimeter}(S)^2}$	R	
convexity	ZUNIC & ROSIN (2004)	$\frac{\text{Area}(S)}{\text{Area}(\text{CH}(S))}$	SAGA GIS , R	Convex Hull
length-width-ratio	YI & MARSHALL (2000)	$\frac{\text{LonVec}(S)}{\text{SmVec}(S)}$	R	LengthWidthRatio (s. 4.3.4)
object orientation	YI & MARSHALL (2000)	$\text{atan2}(\text{LonVec}(S))$	R	MainDirection (s. 4.3.4)

WS: window size. SR: search radius. Dst: distance. D: directions. S: shape/polygon. CH: convex hull. LonVec/SmVec: longer/smaller vector. If no settings set then default parameters.

#### 4.1.2.1 Segmentation Level 1: Detection of Scarp

The detection of the scarps is the crucial task of the detection of the whole landslide, since the result served as input for the subsequent segmentations levels.



**Figure 4-4: Concept of segmentation level 1 (own representation).**

As the size of the landslide can widely differ, also the scarp characteristics are not identical in any landslide. Therefore, in L1 the segmentation was computed with a small-scale parameter resulting in a fine-scaled output to receive multiple parts of one scarp. The objective in L1 was then to merge these parts into a potential scarp based on suitable selections (Figure 4-4).

Fine-scale segmentations can result in a huge number of objects. However, it is assumed that landslide scarps are located near areas characterized by convex slope breaks, or ridges. Those ridges can indicate a landslide crown, and are related to surface convexity (TAROLLI et al. 2012:71). TAROLLI et al. (2012) examined the detection of landslide crowns and bank erosion using the maximum curvature with different window sizes (s. WOOD 1996) and statistically calculated thresholds. The study was then revisited by LIN et al. (2013) and successfully tested with different data in another study area.

Therefore, to restrict the area where potential landslide scarps could be located, a pre-selection of the segmentation objects took place based on the work of TAROLLI et al. (2012) and LIN et al. (2013). The selection of the maximum curvature ( $C_{max}$ ) cells indicating a crown was calculated by the following expression (TAROLLI et al. 2012:72):

$$C_{max} > m * IQR_{C_{max}}$$

$IQR$  stands for interquartile range - a measurement of statistical dispersion, and was calculated as the difference between the third and first quartiles. The selection threshold was then calculated by taking  $m$ -times the  $IQR$  of  $C_{max}$ . Afterwards, the selection was buffered by a suitable buffer size to determine the area where landslide scarps were likely to be located.

The morphometric characteristics of the scarps are shown in Table 4-1. For the detection, all land surface parameters and shape metrics were investigated. However, the following parameters were especially suitable for the corresponding characteristics:

- steepness: slope (s. 2.1.2.3),
- semi-circularity and concavity in planform: convexity (s. 2.1.3),
- main direction  $\perp$  flow direction: flow direction (s. 2.1.2.4), and main direction (s. 2.1.3),
- false positive candidates: flow accumulation (s. 2.1.2.4), openness (s. 2.1.2.6) and sky-view factor (s. 2.1.2.5).

The characteristics of high surface roughness in downslope area was investigated in segmentation level 3.

#### 4.1.2.2 Segmentation Level 2: Detection of Flank

The detection of the flanks is a challenging task. Besides their morphometric characteristics, also their position to the scarp and the landslide body has to be considered. If a flank exists, the detection should be easier upslope near the scarp than downslope where flanks are less clear (VAN DEN EECKHAUT et al. 2012:33ff.).

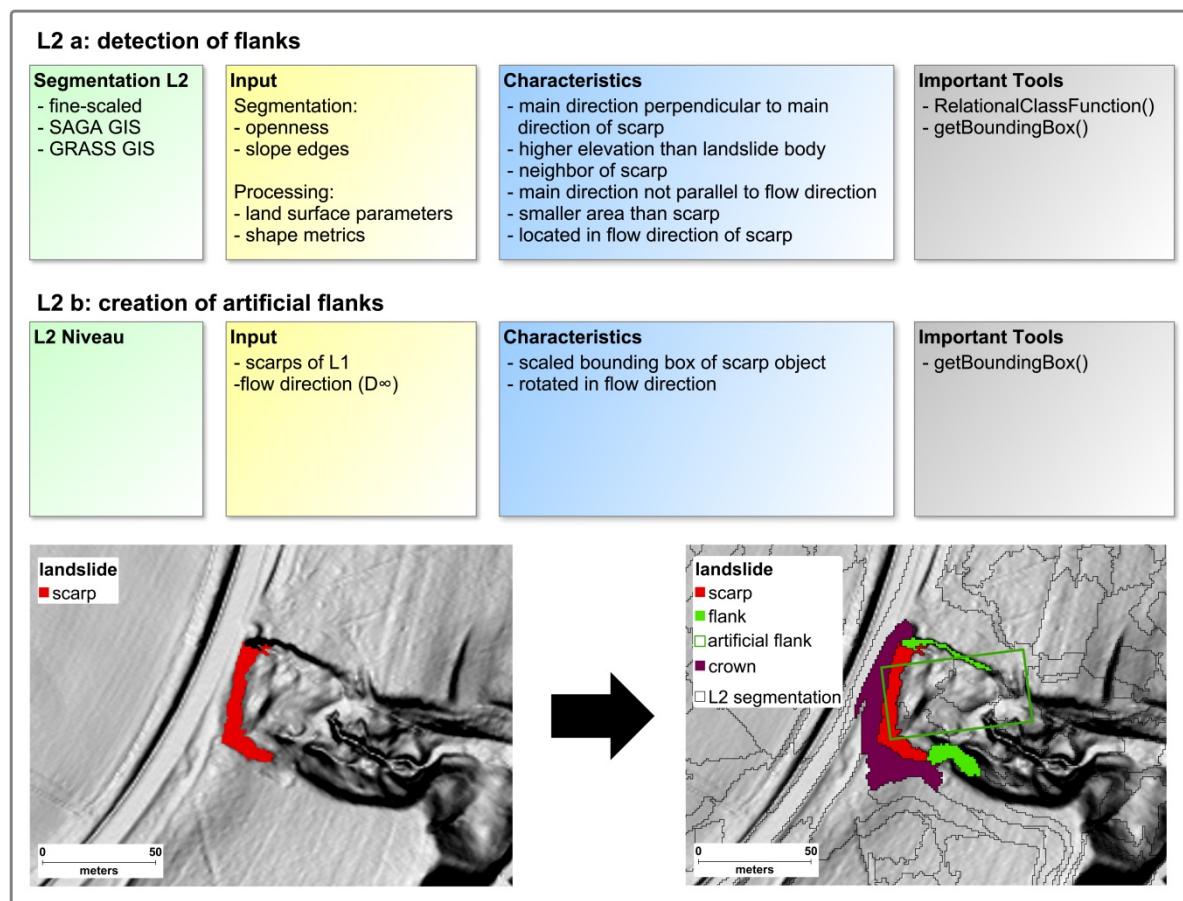


Figure 4-5: Concept of segmentation level 2 (OWN REPRESENTATION).

In the frame of this thesis, besides the morphometric characteristics and object relations (Table 4-1) the following additional assumptions for the detection of flanks were considered:

- flank objects are direct neighbors of scarp objects,
- flanks have a certain steepness (they are not flat),
- flanks are located in flow direction of a scarp,
- the main direction of a flank is not parallel to its flow direction,
- flanks have a smaller area than the weighted mean area of their belonging scarps.

To detect small morphometric changes on the surface, the segmentation must be adapted accordingly. Hence, in L2 the segmentation was computed with a small-scale parameter resulting in a fine-scaled output (Figure 4-5).

To reduce the large number of objects resulting from a fine-scaled segmentation and to restrict the location where potential flanks could be located, the resulted scarps of L1 were buffered and then intersected with the L2 segmentation.

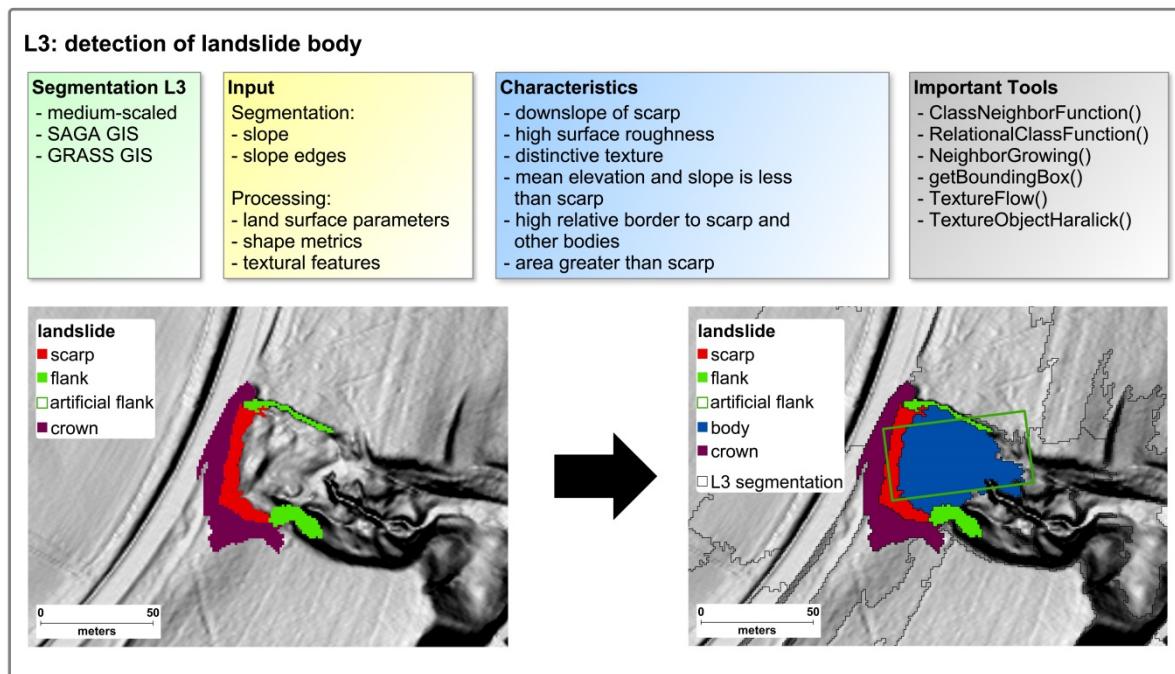
In the following, land surface parameters, shape metrics and object relations that were found to be suitable for the classification, are listed:

- main direction  $\perp$  main direction of scarp: main direction (s. 2.1.3), *RelationalClassFunction* (s. 4.3.5),
- steepness: slope (s. 2.1.2.3),
- location in flow direction: bounding box in flow direction (s. 4.3.3),
- main direction  $\parallel$  flow direction: flow direction (s. 2.1.2.4), and main direction,
- area of flanks  $\leq$  weighted mean area of scarps: *RelationalClassFunction*,
- false positive candidates: openness (s. 2.1.2.6), maximum curvature (s. 2.1.2.3), shape index and maximum distance-area ratio (s. 2.1.3).

The comparison of the elevation of the upslope part to the landslide body could not be investigated on this level.

#### 4.1.2.3 Segmentation Level 3: Detection of Landslide Body and Cleaning

The correct identification of the landslide body is important to detect landslides in its entirety, and to unselect possible false-positive scarps. Generally, a landslide body is located downslope of a scarp and in the optimal case bordered by flanks. In addition, it is characterized by a hummocky topography represented by high surface roughness values (Table 4-1).



**Figure 4-6: Concept of segmentation level 3 (own representation).**

Besides their morphometrical and positional characteristics, additional assumptions for the detection of landslide bodies were considered:

- the slope of a body is less than its belonging scarp,
- the mean elevation is less than its belonging scarp,
- for small scarps the area ratio to the body should not exceed a certain threshold,
- the body should have a greater area than its belonging scarp,
- distinctive texture in flow direction,
- body parts should not have a high shape index or a large length-width-ratio,
- body parts should have a high relative border to its belonging scarp and other body parts,
- body parts are not too steep as well as not too flat.

As mentioned in chapter 3.3 the area of the landslides in the study area varies widely (s. Table 3-2). The segmentation in L3 must therefore be capable to create small as well as large objects. Hence, a medium-scale segmentation was computed with a greater scale parameter than in L1 or L2 (Figure 4-6).

To delimit the objects to areas where potential bodies could be located, the L3 segmentation was intersected with the buffered result of L1 and L2. Moreover, in L3 object relationships between potential scarps and bodies were computed and considered in order to remove possible false-positive candidates.

Besides land surface parameters, shape metrics and object relations, also textural features were investigated in this level. In the following, suitable separability criteria for the classification of the landslide body as well as for the removal of false-positive scarp candidates are listed:

- location in flow direction: bounding box in flow direction (s. 4.3.3),
- steepness: slope (s. 2.1.2.3),
- relation to scarp: relative border, ratio, weighted-mean and weighted-mean difference (*RelationalClassFunction*, s. 4.3.5),
- relation to other bodies: relative border (*RelationalClassFunction*),
- shape of body: shape index and length-width-ratio (s. 2.1.3)
- texture<sub>Flow</sub>: entropy, contrast, variance and IDM (s. 2.1.4 and 4.3.6),
- false positive candidates: relation between scarp and body, surface roughness (s. 2.1.2.7), sky-view factor (s. 2.1.2.5), slope, shape metrics (s. 2.1.3) and entropy.

At the end of the classification step by the *NeighborGrowing* function (s. 4.3.5) all landslide parts (scarp, flank and body) were grown together starting from the landslide scarp.

#### 4.1.3 K-Means Thresholding

A suitable classification method is crucial to successfully classify the different landslide parts and to distinguish them from possible false positive candidates. Commonly, after a segmentation procedure the user selects some representative sample objects that correspond to a specific class as training set for a classifier. Then, by supervised or unsupervised statistical approaches (or classifier) such as Random Forest (STUMPF & KERLE 2011) or supervised support vector machines (VAN DEN EECKHAUT et al. 2012) the objects are classified. However, these semi-automatic approaches have the limitation to be dependent on the quality of the subjective selection of training data and its sample size (SANTOS et al. 2013).

To avoid the interaction with the objects the k-means thresholding method of MARTHA et al. (2011) was adapted for this thesis. This method calculates data-driven thresholds by cluster analysis with k-means (LIU & YU 2009, MARTHA et al. 2011:4932). K-means cluster analysis finds natural homogeneous groups in continuous data. Every group has a center or centroid that, according to MARTHA et al. (2011), can be used as a threshold to identify landslide candidates (MARTHA et al. 2011:4932f.). However, the k-means algorithm has the drawback that the number of clusters have to be determined. To overcome this limitation MARTHA et al. (2011) suggested to automatically select the number of clusters by using a Schwarz Bayesian criterion (MARTHA et al. 2011:4932f., SCHWARZ 1978). K-means thresholding has the advantage to handle large data sets, to be transferable to other areas, and to be applicable on different types of Earth observation data (MARTHA et al. 2011:4932f.).

In the frame of this thesis, the k-means thresholding approach was implemented by the *kMeansThresholds* function (s. 4.3.7).

#### 4.1.4 Accuracy Assessment

The accuracy assessment of a classified, thematic map is essential to quantify the quality of the classification. In addition, by that third persons are able to adequately use the product for further applications (STEHMAN & CZAPLEWSKI 1998:331). The right selection of the sampling design and the evaluation method is important for the meaningfulness of the assessment.

In order to access the accuracy of the classification independent of the inventory, every classified object was checked individually and evaluated by the author. The classification accuracy was then obtained from the ratio of the number of correctly and incorrectly classified objects. In the special case, that there was no scarp or minor-scarp located in a landslide area, but the associated body, then this landslide was attributed as incorrect.

To avoid bias by wrongly grown landslides into landslides of the inventory, for the accuracy assessment based on the available landslide inventory, only the as correctly classified landslides and landslide parts were used. However, the accuracy assessment of landslide classifications based on landslide inventories and object-oriented image analysis is still a challenging task and difficult to perform (VAN DEN EECKHAUT et al. 2012:37). According to VAN DEN EECKHAUT et al. (2012), this can be derived from the following two reasons: 1. there are no completely perfect and objectively created landslide inventory maps for the benchmark, and 2. no standard evaluation method to assess the quality of an image segmentation or even a multiresolution segmentation exists (VAN DEN EECKHAUT et al. 2012:37).

In this study the accuracy assessment based on the landslide inventory was adapted from VAN DEN EECKHAUT et al. (2012).

VAN DEN EECKHAUT et al. (2012) suggested two on landslide inventory based measurements. The first method is a comparison between the number of scarps and landslides that are correctly identified or missed. Here, a landslide is considered as correctly classified when the scarp together with at least  $x\%$  of the landslide body was detected. In their study VAN DEN EECKHAUT et al. (2012) used  $x = 50$  as threshold (VAN DEN EECKHAUT et al. 2012:37).

The second method is a calculation of the position mismatch (*PM*) between two inventory maps and is calculated as follows:

$$PM = \frac{A_{EUO} - A_{E\cap O}}{A_{EUO}} * 100$$

With  $\cup$  as symbol for union and  $\cap$  for intersection, respectively. Thus,  $A_{EUO}$  represents the area of a landslide (scarp, flank and body) either on the expert-based or on the classification-based inventory, and  $A_{E\cap O}$  represents the landslide area of both inventories. The values of

*PM* range from 0% for an exact match to 100% for no match between the inventories (VAN DEN EECKHAUT et al. 2012:38).

## 4.2 GIS Open Source Software

In the last decade, free and open source software (FOSS) has experienced great popularity in all scientific domains. In the context of this thesis, the term “free and open source” encompasses software that are released under the license agreement of the Free Software Foundation (FSF) as well as the Open Source Initiative (OSI) (for further details see STEINIGER & BOCHER 2009:1347ff.). Especially, in GIS science with the establishment of the Open Source Geospatial Foundation (OSGeo) in 2006, free software and algorithms are increasingly published under open source licenses (STEINIGER & BOCHER 2009:1346ff.).

There are more than 300 free GIS software projects (STEINIGER & BOCHER 2009:1346). However, here only GIS software packages relating to geomorphometric analysis such as GRASS, SAGA, ILWIS, LandSerf, MicroDEM, TauDEM, TAS, or QGIS, were of interest (WOOD 2009:257f.). Moreover, the possibility to interconnect this software with the development environment R was an important decision criterion. Hence, for basic geomorphometric operations GRASS and SAGA GIS were chosen. In addition, software packages with specialized algorithms such as TauDEM or Relief Visualization Toolbox were used.

### 4.2.1 R

The R system is a language and a free development environment for data analysis and graphics (R CORE TEAM 2016). It originated from the two languages S and Scheme, however, code and libraries written in C, C++, Fortran, or Java can easily be linked to R (BIVAND et al. 2013:2, R CORE TEAM 2016). The major strengths of R are the manifold standard and innovative statistical analysis methods, and the easy possibility to share and extend R functionalities by user programmed packages (BIVAND et al. 2013:2f., VENABLES et al. 2016:2).

Many packages for handling and analyzing spatial data were contributed over the last 15 years by several authors (BIVAND et al. 2013:3). A fundamental overview and introduction is given by BIVAND (2011) and BIVAND et al. (2013). In addition, meanwhile R has even the capability to include the entire functionalities of open source desktop GIS projects such as GRASS, SAGA, or QGIS by packages (*spgrass6/rgrass7*, *RSAGA*, *RQGIS*). For this thesis, the R version 3.3.2 was used.

### 4.2.2 GRASS GIS

GRASS - Geographic Resources Analysis Support System, is a GIS software combined with integrated image processing and data visualization subsystems (NETELER & MITASOVA 2008:3). It has its roots already in the early 1980s when the US Army Construction Engineering Research Lab (CERL) started to develop an alternative to ESRI's ArcInfo (STEINIGER

& BOCHER 2009:1354). Since the adaption of the GNU General Public License (GNU GPL) in 1999, GRASS embraces the open source philosophy and records an increasing user and developer community (NETELER & MITASOVA 2008:3, STEINIGER & BOCHER 2009:1354). The actual version consists of more than 350 modules for management, processing, modeling, and visualization of geospatial data, as well as for graphics and maps production (NETELER & MITASOVA 2008:3).

For this thesis, the stable release GRASS GIS 7.2.0 was used and combined with R by the *rgrass7* package of BIVAND (2016).

#### 4.2.3 SAGA GIS

SAGA stands for System for Automated Geoscientific Analyses, and its idea evolved in the late 1990s during research and development projects of the department for Physical Geography at the university of Gottingen, Germany (CONRAD et al. 2015:1991). SAGA emerged from three applications having the major focus on digital terrain analysis: SARA (System zur Automatischen Reliefanalyse), SADO (System zur Analyse und Diskretisierung von Oberflächen), and DiDeM (Programm für Digitale Gelände-Modellierung) (CONRAD et al. 2015:1992, 1996). SAGAs first public release was in 2004. Since then it developed to a well-known FOSS GIS platform with a growing user community and many developers (CONRAD et al. 2015:2002).

Nowadays, there are more than 600 methods encompassing the entire spectrum of contemporary GIS, such as raster and vector analysis, data management and conversion tools, mapping, or spatial and geostatistic functionalities. However, its focus and strengths still remain on terrain analysis (CONRAD et al. 2015:1991f.).

Based on the work of BRENNING (2008) SAGA can be integrated into the R environment by the *RSAGA* package (BRENNING 2008, CONRAD et al. 2015:1995). For this thesis, the SAGA GIS 2.2.2 version was used.

#### 4.2.4 Additional Software

*TauDEM*. Most GIS software contains hydrological modeling tools, such as flow direction and accumulation estimation, watershed delineation, or channel network extraction. Depending on the implemented algorithm, their result can widely differ. Recently, there is no tool available neither in GRASS nor in SAGA GIS to simply compute the  $D^\infty$  flow direction raster based on the method of TARBOTON (1997). Therefore, the free software TauDEM (Terrain Analysis Using Digital Elevation Models) in the version 5.3 including the  $D^\infty$  algorithm was installed and implemented in R by *system()*-commands. TauDEM has its focus on hydrological analysis of topography and was first released in 2001 (TARBOTON et al. 2015).

*Relief Visualization Toolbox.* The sky-view factor is implemented in the stand-alone Relief Visualization Toolbox (RVT, recent version 1.3, first release November 2013), but can also be accessed through the GRASS GIS add-on *r.skyview*. However, the toolbox performs around 97 times faster than the add-on with respect to the author's computer configuration. The RVT can be easily integrated in R through *system()*-commands. However, RVT is coded in IDL (Interactive Data Language) and based on IDL run-time license. When the tool starts, a confirm message window of the IDL Virtual Machine pops up. This circumstance is for an automated implementation disturbing. To circumvent the confirmation step, an additional .bat-file including the *sendKeys.bat* of NPOCMAKA (2015) was used to automatically "push enter". In the frame of this thesis, a function was implemented to easily access the RVT in R (*skyViewFactor*, s. 4.3.7).

### 4.3 Implementation of Functions in R

Despite numerous functions and modules of the open source products, there were still some tools missing to fulfill the objective of this thesis. In this chapter, the functions coded for the landslide detections by the author are briefly explained. Core functions are visually supported by flow charts showing a simplified workflow. The references to the source code, the *Lslide*-package, and the R scripts for the detection can be found in chapter 1.

There are many R packages specific for geo-science applications besides the already mentioned desktop GIS packages (s. 4.2.2 and 4.2.3). In this thesis the following packages were found to be very valuable: *maptools* (BIVAND & LEWIN-KOH 2016), *raster* (HIJMANS 2016), *rgeos* (BIVAND & RUNDEL), *sf* (PEBESMA 2017), *sp* (PEBESMA & BIVAND 2005), *spdep* (BIVAND & PIRAS 2015), and *shapefile* (STABLER 2013). Especially the *spdep* and *sf* package greatly contributed to the development of this thesis. The former enables the possibility to create neighborhoods between objects, and by the latter it is possible to easily handle large shapefiles even in a performant way.

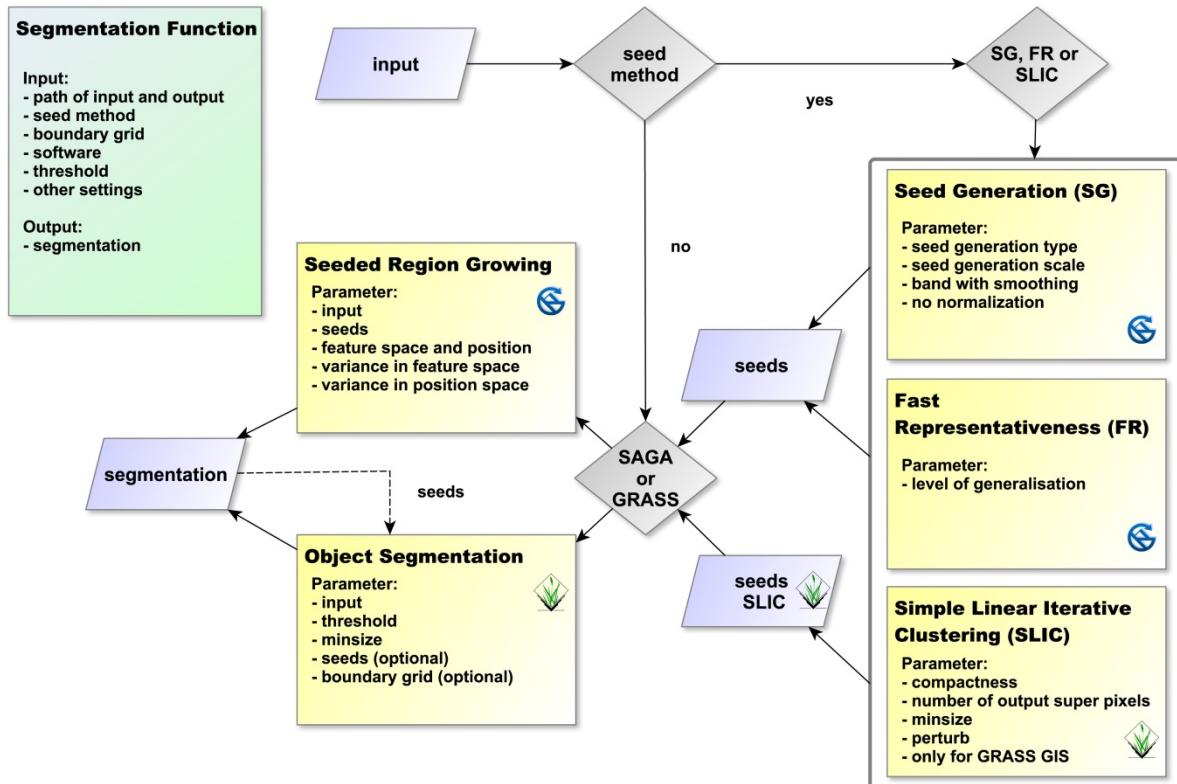
Besides, for all computationally intensive functions the run-time was measured by using the default settings of the function. Therefore, all mentioned time measurements are related to the author's laptop configuration: dual-core Intel® Core™ i7 CPU M620 2.67 GHz, 8 Giga-byte RAM and Windows 10 64-bit system.

In the following the meaning of the shapes and symbols used in the following flow charts are explained:

**Table 4-3: Explanation of flow chart symbols (OWN REPRESENTATION).**

shape	description	shape	description
	function name function parameters input and output files		process by a function or module
	input/output of data during process		decision/condition
	GRASS GIS module		SAGA GIS module
	R functionalities		control pass to next element
			control pass after decision/condition of a loop

#### 4.3.1 Segmentation Function



For legend see Table 4-3.

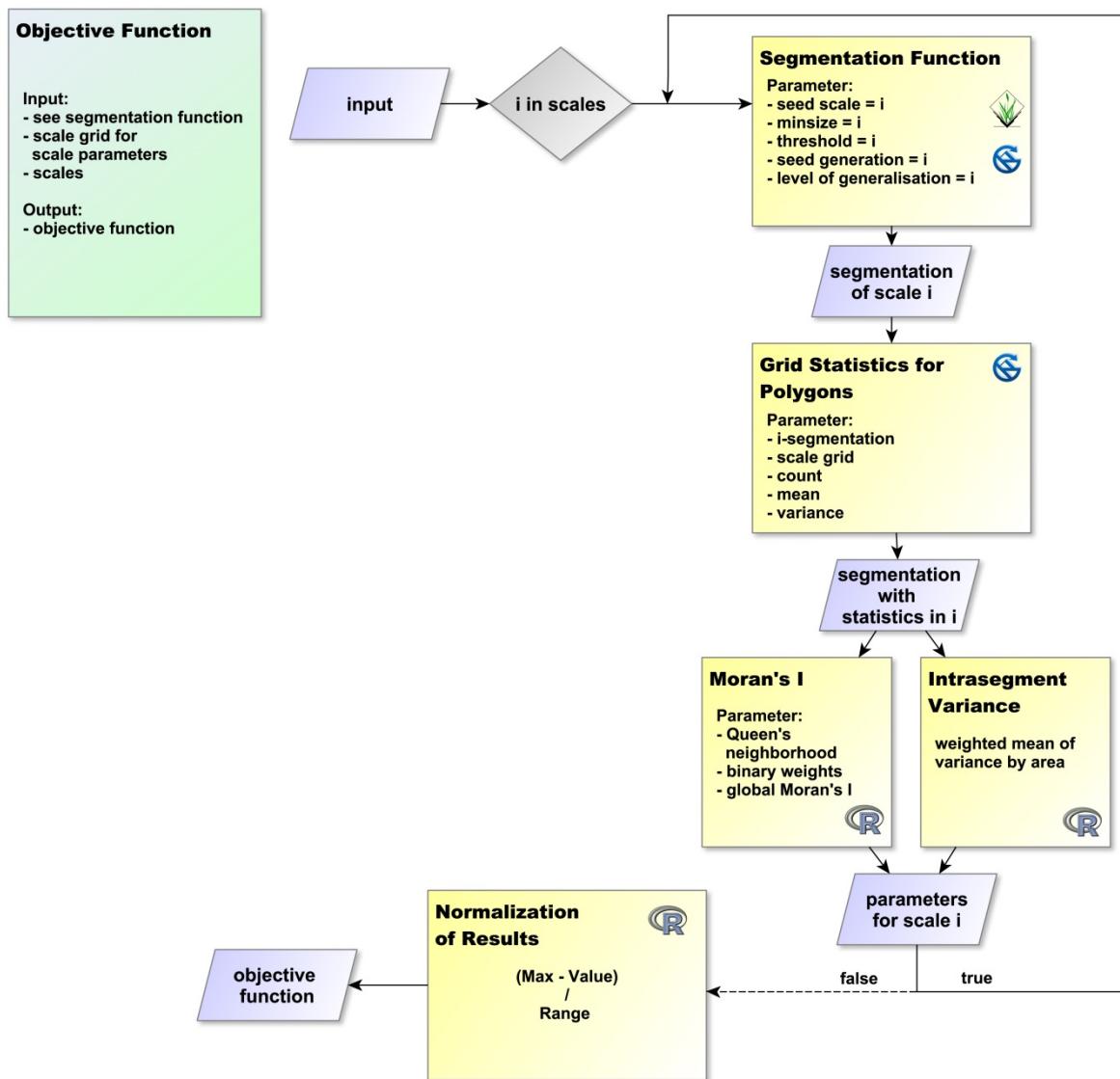
**Figure 4-7: Flow chart of segmentation function (OWN REPRESENTATION).**

The segmentation function is the core function for the object-oriented landslide detection (Figure 4-7). The function takes raster files as input, and segments the data into homogene-

ous regions or objects depending on the settings (c.f. 4.1.1.2). Restricted to the software used in this function, the segmentation is performed either as seeded region growing, or region growing algorithm, or as simple linear iterative clustering. The segmented output is given in raster and shapefile format for further analysis.

The segmentation is computationally very demanding. A fine-scale segmentation for the whole study area takes in SAGA GIS around 80 minutes and in GRASS GIS up to 700 minutes, respectively.

#### 4.3.2 Objective Function



For legend see Table 4-3.

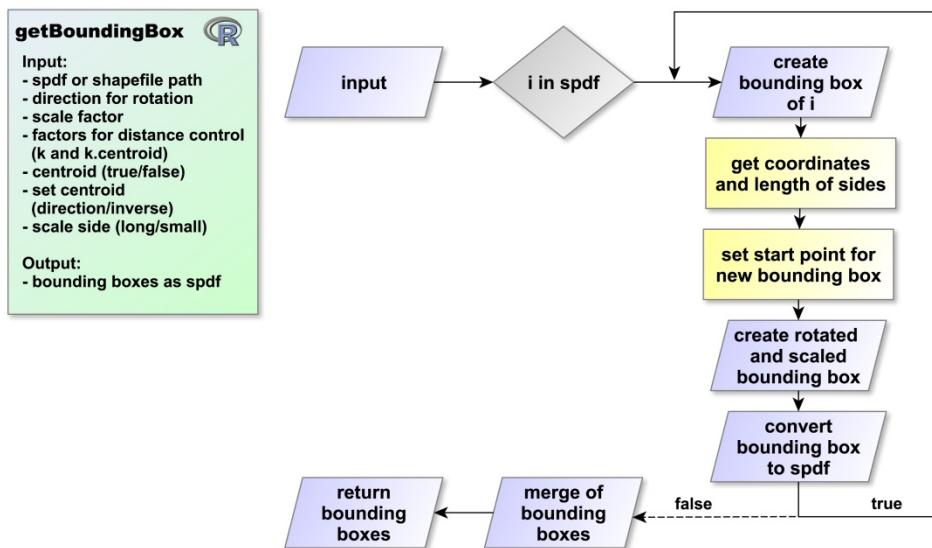
Figure 4-8: Flow chart of objective function (OWN REPRESENTATION).

The *objective function* provides statistics for estimating optimal scale parameters, and its theoretical background is explained in more details in chapter 4.1.1.4. In the function, the segmentation steps as well as the computation of the statistics (Moran's I and intersegment

variance) are iteratively performed in dependence on the scale input. After the loop, the statistics are normalized and a data table containing the result is returned (Figure 4-8).

The *objective function* is based on the *segmentation function* and has therefore the same minimum overhead. The run-time in this implementation is linear depending on the amount of scales that are iterated in the for-loop. However, further prospective can be to parallelize the process to speed up the computation.

### 4.3.3 Bounding Box in Flow Direction



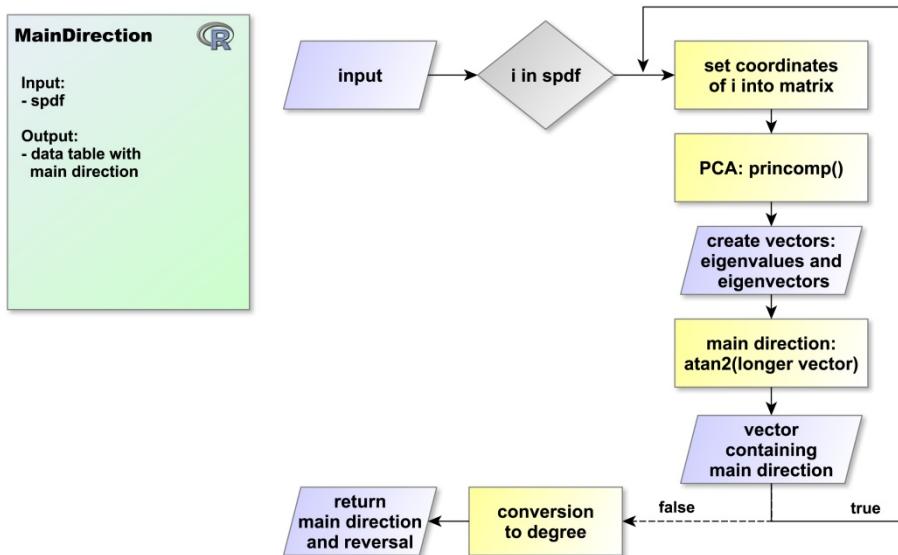
For legend see Table 4-3.

Figure 4-9: Flow chart of `getBoundingBox` function (OWN REPRESENTATION).

The function `getBoundingBox` has the objective to create a bounding box for given objects and to rotate and scale it into a given direction, for example the flow direction. Relating to landslide detection, the idea is to identify objects that lay in a specific direction of another, for example a landslide body situated in flow direction to a landslide scarp. Furthermore, the function enables the possibility to choose which side (long/small) of a given object shall be scaled in flow direction. The scale factors can also be adapted adequately. In addition, there is the option to set the starting point of the rotated and scaled bounding box relative to the centroid of the initial bounding box. This is particularly helpful when an object has a very low convexity value. Here, the bounding box starting point can be set in flow direction to avoid intersections of polygons in inverse flow direction.

The function is completely programmed with R functionalities (Figure 4-9). The computation of bounding boxes for 3,220 objects was done in circa 17.4 seconds.

#### 4.3.4 Object Orientation



For legend see Table 4-3.

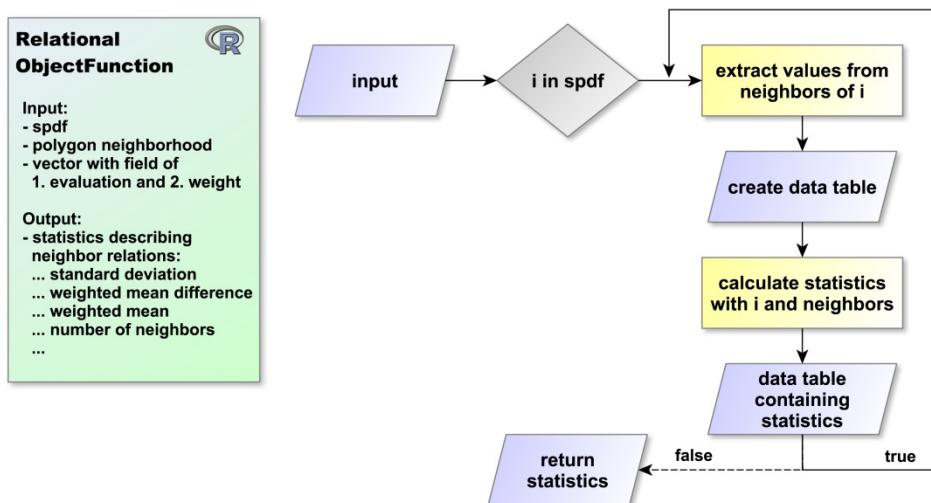
Figure 4-10: Flow chart of `MainDirection` function (own representation).

The functions `MainDirection` and `LengthWidthRatio` are principally based on the same algorithm, and therefore only the `MainDirection` is shown in Figure 4-10. The theory is briefly explained in chapter 2.1.3. The implementation of the function is completely done by R functionalities. Here, the principal component analysis is done by the `princomp()` function.

The `MainDirection` executed the calculation for 23,170 objects in 2.62 minutes. The function `LengthWidthRatio` finished for the same input in 1.56 minutes.

#### 4.3.5 Neighbor Operations

Information on the neighborhood of objects is essential to perform an object-oriented analysis. In this study, four functions with different purpose were created based on the object neighborhood: `RelationalObjectFunction`, `RelationalClassFunction`, `ClassNeighborFunction`, and `NeighborGrowing`. The first three functions are computationally similar, and, thus, only the `RelationalObjectFunction` is shown in Figure 4-11. All four functions are implemented by using R functionalities only.



For legend see Table 4-3.

**Figure 4-11: Flow chart of RelationalObjectFunction (OWN REPRESENTATION).**

The computation of *RelationalObjectFunction*, *RelationalClassFunction*, and *ClassNeighborFunction* is based on the principle of the relational features and class-related features in TRIMBLE DOCUMENTATION (2014:253-255; 375–380). Thus, based on a given field of evaluation and weight, for every object the following, for this thesis relevant, statistics of its neighbors are calculated: “weighted mean”, “standard deviation”, “weighted mean difference to”, “relative border to class”, “absolute border to class”, and “distance to class”.

The *RelationalObjectFunction* computes neighborhood statistics iteratively for every object.

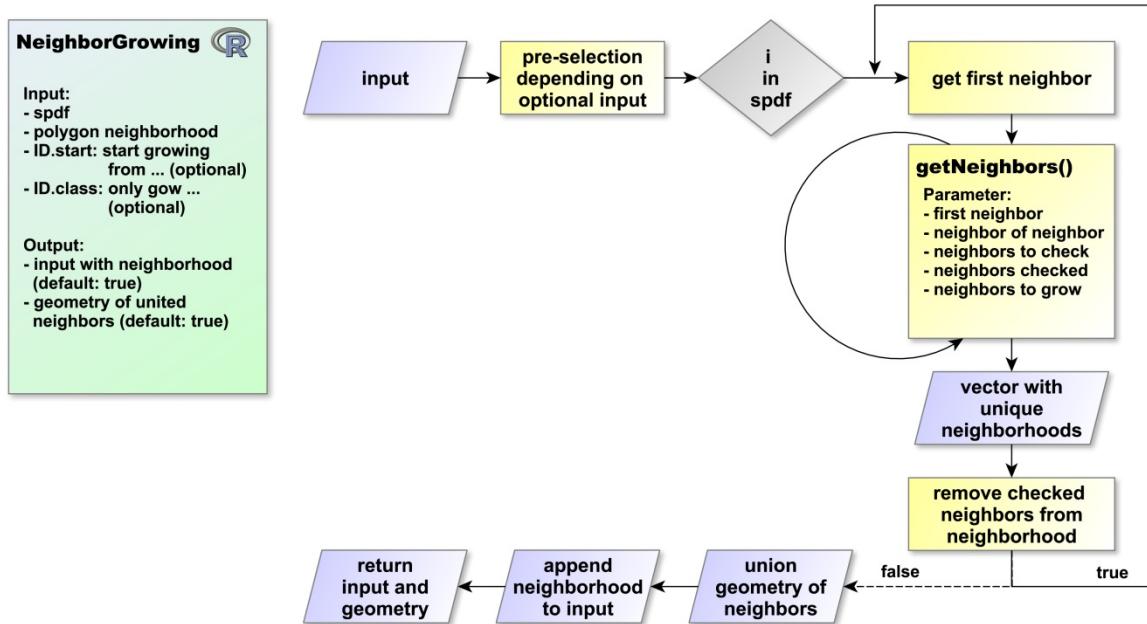
The *RelationalClassFunction* calculate statistics from neighbors of a given class to the given class. Moreover, it is possible to calculate statistics for neighbors in a specific direction of a class object by setting the parameter for the *getBoundingBox* in the function call. In addition, if the neighbors are attributed to another second class, also a class-to-class relationship can be performed. Furthermore, attributing the first and the second class must not be necessarily hard coded, instead it is also possible to use expressions for the class-to-class relationship. However, here it can happen that some objects are assigned to both classes. As default, these objects are excluded from the statistics, but they can be included by setting the specific argument (*class.as.neighbors*). Moreover, the function is also capable to compute statistics on angle inputs by setting the function argument of *var.is.angle* to true.

The *ClassNeighborFunction* computes statistics for every object of a specific class to its neighbors. It is computationally identical to the *RelationalObjectFunction*. However, the function additionally enables the possibility to perform statistics for neighbors in a specific direction likewise the *RelationalClassFunction*, and to select class objects by expressions.

The run-time of the *RelationalObjectFunction* was 5.03 minutes for 23,170 objects with 115,166 neighbors. The *RelationalClassFunction* executed 12,858 neighbor objects for 3,220

class objects in 12.46 minutes. The *ClassNeighborFunction* took around 1.10 minutes to compute 3,220 class and its 12,858 neighbor objects. The longer run-time of the *RelationalClassFunction* can be ascribed to the geometrical intersection for getting the information about the relative border and distance to the selected class(es).

Besides the computation of statistics, the object neighborhood is also valuable to grow or merge neighbors together into one single object (Figure 4-12).



For legend see Table 4-3.

Figure 4-12: Flow chart of NeighborGrowing function (OWN REPRESENTATION).

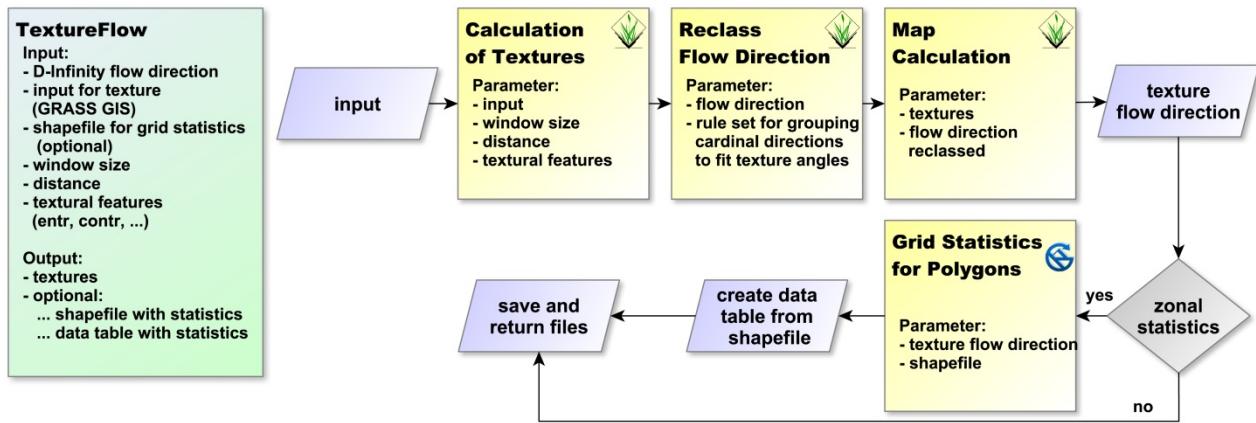
The function *NeighborGrowing* merges neighbors to one single object. The procedure is done by a combination of a for-loop and the recursive function *getNeighbors*. Iterating through the objects, the first neighbor is taken in the first step. Then, the recursive call of *getNeighbors* gets the neighbors of the first neighbor and so on until all possible related neighbors are found and checked. Neighbors belonging together are stored in a vector and are finally removed from the neighborhood to avoid a continuous loop. As default, the merged geometry of neighbors and a column appended to the input, specifying those belonging neighbors by increasing numbers, are returned. Moreover, it is possible to grow only specific class objects or to start growing only from specific objects.

Related to the objective of this thesis, this function was especially useful to grow landslide parts to an entire final landslide object, as well as for all growing in the sub-steps (scars, flanks and crowns).

For the growing of 8,730 objects into 4,099 objects the run-time was around 1.39 minutes.

### 4.3.6 Texture in Flow Direction

The importance of the computation of the texture in flow direction is described in chapter 2.1.4. The implementation of the *TextureFlow* function is mainly based on the modules of SAGA and GRASS GIS (Figure 4-13).



For legend see Table 4-3.

Figure 4-13: Flow chart of *TextureFlow* function (OWN REPRESENTATION).

The input for this function is the D-Infinity flow direction raster, a raster on which the texture shall be calculated (here slope), and the texture settings (window size, distance, textural features). As default, the texture measurements of contrast, correlation, variance, sum average, and entropy are calculated for all angles ( $0^\circ$ ,  $45^\circ$ ,  $90^\circ$ ,  $135^\circ$ ). Then, the flow directions are grouped to cardinal directions to fit the textural angles. Subsequently, based on conditions of the re-classed flow direction, the texture is computed in flow direction. Optionally, zonal statistics can be further calculated for a given shapefile input and returned as data table.

As default, the output of *TextureFlow* is the texture in flow direction and the texture perpendicular to the flow direction. As further perspective, the computation of the ratio of both outputs can be implemented.

The computation of *TextureFlow* took around 51 minutes for the entry study area with a window size of 5 m and distance of 1 m. Here, an investigation of the performance of the textural features in the *glcm* package of ZVOLEFF (2016) can be a future perspective.

### 4.3.7 Other Functions

The following functions are auxiliary functions, or modified and adapted functions from other R packages. Therefore, they are not explained in detail but can be examined in the *Lslide* package.

*calcAccumulationBuffer*. This function enables the possibility to automatically create a stream network by buffering thresholds of a flow accumulation input. Here, the buffer sizes can be adapted related to given accumulation thresholds.

*checkGeometry*. The *checkGeometry* function checks spatial data for invalid geometries. If an invalid geometry is detected, it will be repaired by the *clgeo\_Clean* function of the *clean-geo* package (BLONDEL 2016).

*correctDBF*. The *correctDBF* function loads the .dbf-file of a shapefile, changes the header to a given header, and overwrites the old .dbf-file with a .dbf-file containing the new header. This function is especially useful for shapefile-outputs of SAGA GIS. Here, often the column names are not in the ESRI shapefile format.

*kMeansThresholds*. The function *kMeansThresholds* computes cluster centers for a given data frame input by the *kmeans* function of the basic R *stats* package. The number of cluster can be set manually, however as default, they are estimated by the Schwarz Bayesian criterion using the *Mclust* function of the *mclust* package (FRALEY & RAFTERY 2002). Big data input can result in a long run-time. Therefore, options are included to randomly subset and sample the data input. The run-time for 23,170 objects was around 4.3 minutes.

*ObjectFeatures*. This function is an adapted version of the *computeFeatures.shape* function in the *EBImage* package (PAU et al. 2010). It is insofar modified, that the segmentation output can fit into *computeFeatures.shape*, and that the function return is a suitable data table for further analysis.

*replaceInvalids*. This function simply loops through every column of a given input data frame and replaces invalid values (*NA*, *NAN* and *NULL*) by a new user-defined value (default: -9999). When the data frame of spatial data has such invalid values, after exporting as shapefile, there is the great possibility that some standard desktop GIS software are not able to read the .dbf-file. By using a no-data value, this failure should be eradicated.

*skyViewFactor*. With the *skyViewFactor* function it is possible to directly address the stand-alone Relief Visualization Toolbox of ZAKŠEK et al. (2011) by *system()*-commands (s. 4.2.4).

*TextureObjectHaralick*. This function is, like *ObjectFeatures*, a modified and adapted version of a function in the *EBImage* package - *computeFeatures.haralick*. The function is very important for object-oriented analysis, because it enables the possibility to calculate textures for every object using the values inside of an object. Therefore, the calculation is similar to that of the Definiens eCognition software. The *TextureObjectHaralick* is insofar modified that the inputs can be raster objects. In addition, the raster for the texture is linear normalized to grayscale values (0-255) to fit into *computeFeatures.haralick*. For computing textural features for 23,170 objects, the function needs 3.05 minutes run-time.

## 5 Results

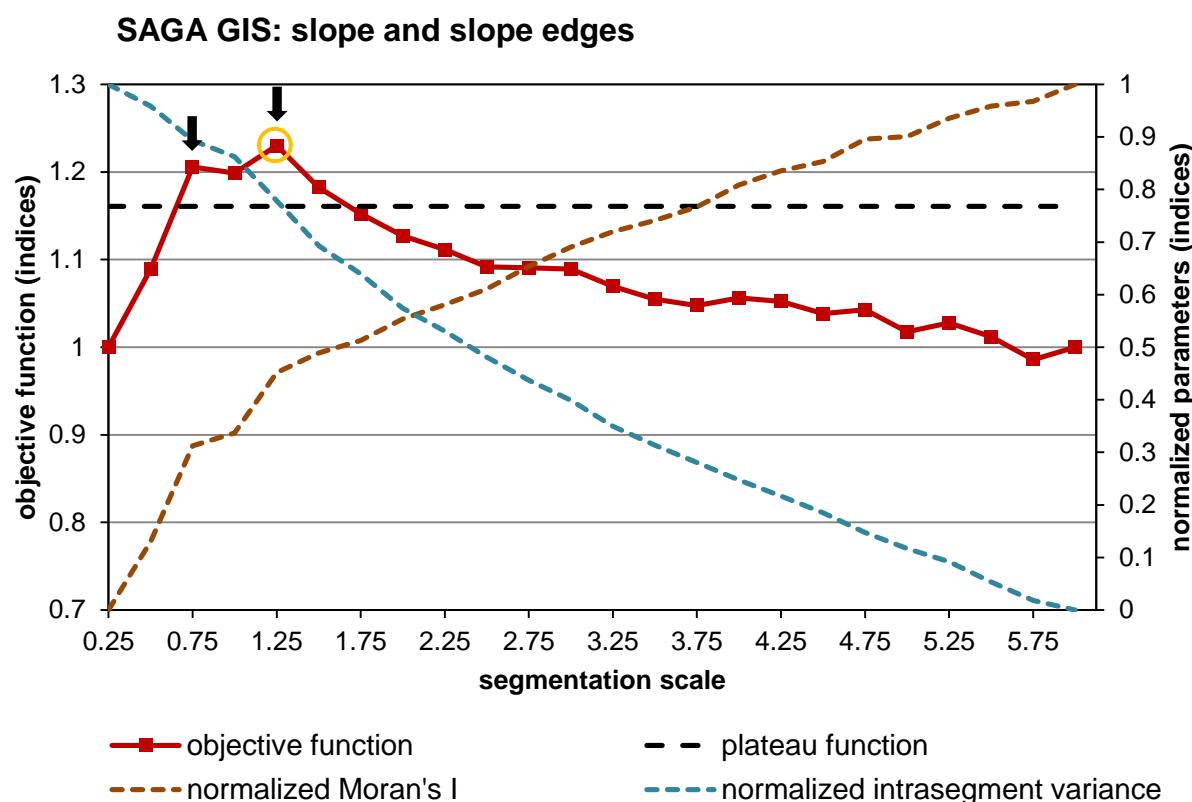
### 5.1 Landslide Detection

The results of the landslide detection are divided into the three different segmentation levels. In each subchapter, at first, the result of the corresponding objective function is shown, then the k-means thresholds are tabularized, and finally the detection based on the segmentation, thresholds and queries is performed.

In each segmentation level, outputs are created and objects classified. The Table A-1 in the Appendix A gives an overview of the outputs and created classes during the processes.

#### 5.1.1 Segmentation Level 1

Segmentation level 1 had the objective to detect potential landslide scarps.



Black arrows: optimal scale parameters. Yellow circle: selected scale.  
Segmentation scale represents the Level of Generalisation in the module Fast Representativeness.

Figure 5-1: Level 1 – Objective function (OWN REPRESENTATION).

In L1 the segmentation scales ranging from 0.25 to 6 were investigated by the objective function (Figure 5-1, Table B-1 in the Appendix B). As segmentation input slope and slope edge were used, and the statistics were calculated based on the slope values.

The graph in Figure 5-1 indicates that the segmentation scales ranging from 0.75 to 1.5 had higher OF values than the plateau function value (1.161). Here, the two peaks at scale 0.75

with the OF value 1.206 and at scale 1.25 with the OF value 1.230 represented the local optimal segmentation scale. For the next processing step, therefore, the segmentation scale with the highest OF value – scale 1.5 – was chosen.

The first segmentation was performed in SAGA GIS using the implemented seeded region growing algorithm. The seeds were computed with the module *Fast Representativeness* and as *Level of Generalisation* the optimal scale parameter of the OF above was used. The segmentation (*L1\_seg.shp*) resulted in 661,210 different objects. To reduce the amount of data only to regions where a potential landslide scarp could be located, a selection was taken based on the analysis of TAROLLI et al. (2012) (s. 4.1.2.1). The IQR threshold for the maximum curvature with  $m = 1.5$  was 0.004074238. Grid cells greater than this threshold were selected, and subsequently buffered with a subjectively set distance of 5 m. This output (*curv\_max15\_IQR\_buf.sdat*) was intersected with *L1\_seg*. Only objects containing at least one cell of *curv\_max15\_IQR\_buf* were further retained. Therefore, the huge amount of segmentation objects could be reduced to 241,007 (*L1\_seg\_sel.shp*).

Based on this intersection the thresholds for the different land surface parameters were computed by k-means thresholding and are tabularized in Table 5-1.

**Table 5-1: Level 1 – k-means thresholds of land surface parameters (own representation).**

<b>Level 1: k-means thresholds</b>					
<i>n</i>	<i>slope</i>	<i>openness</i>	<i>surface roughness</i>	<i>sky-view factor</i>	<i>maximum curvature</i> +
1	3.977664	-5.4716001	0.00102653	0.6446079	-0.0123849
2	8.01043	-2.5721536*	0.00420056*	0.7432842*	-0.00650622
3	12.316246	-0.8806598	0.00840356	0.8167058	-0.00305004
4	17.242991	0.1519328	0.01342047	0.8744564*	-0.00094407
5	23.314519*	0.8253774*	0.01934808	0.9196185	0.00039919
6	30.767841	2.2043472	0.02620597	0.957865	0.00140394
7	39.882225		0.03392244		0.00231077
8			0.04261873		0.00319853
9			0.05398941		0.00411455*
10			0.06819319		0.00515952
11			0.08538878		0.00636943
12			0.1074428		0.00775231
13			0.13743165		0.009351
14			0.18485656		0.01115534
15					0.01320567
16					0.01552164

Level 1: k-means thresholds					
<i>n</i>	<i>slope</i>	<i>openness</i>	<i>surface roughness</i>	<i>sky-view factor</i>	<i>maximum curvature</i> *
17					0.01816075
18					0.02114612
19					0.02465432
20					0.02882207
21					0.03418797
22					0.04179656
23					0.05858715

\* Thresholds calculated for whole dataset. \* Selected threshold for further analysis.

According to the characteristics of a scarp (s. Table 4-1), the objects in *L1\_seg\_sel* were further refined by using suitable thresholds based on the k-means thresholding results (Table 5-1) and their empirical examination by visual inspection of the data. Therefore, the following expression was used to further restrict the objects of a potential landslide scarp (*class 111*):

```
class 111 = slope ≥ 23.314519
& -2.5721536 ≤ openness < 0.8253774
& 0.7432842 ≤ sky view factor < 0.8744564
& 0.004200564 ≤ surface roughness
& area ≥ 25
```

The minimum area of 25 m<sup>2</sup> was empirically set. The seeded region growing algorithm has no options to control the minimum size of an object. Thus, objects even with an area of 1 m<sup>2</sup> can result, especially in noisy regions. Using this area threshold, around 3,500 objects could be additionally removed from *class 111* (Figure 5-2).

In the next step, the focus was on the removing of possible false-positive candidates from *class 111*. Here, objects were excluded that had intersections to following classes:

- *class 44*: intersection of sky-view factor (< 0.7432842) and openness (< -2.5721536),
- *class 55*: maximum curvature ( $\geq 0.00411455$ ),
- *class 66*: buffered flow accumulation depending on thresholds (10,000 | 1 m; 30,000 | 3 m; 70,000 | 7 m; 200,000 | 10 m; 800,000 | 15 m; 1,250,000 | 20 m).

*Class 44* and *class 66* were useful to remove objects in depressions and possible fluvial valleys. By *class 55*, objects that had no neighbors to a possible crown object could be excluded. Thus, finally, *class 111* consisted of 8,730 objects representing possible scarp candidate parts. Then, with the function *NeighborGrowing* every scarp object was merged with its surrounding neighbors to receive one single object per scarp in the optimal case (*L1\_final.shp*).

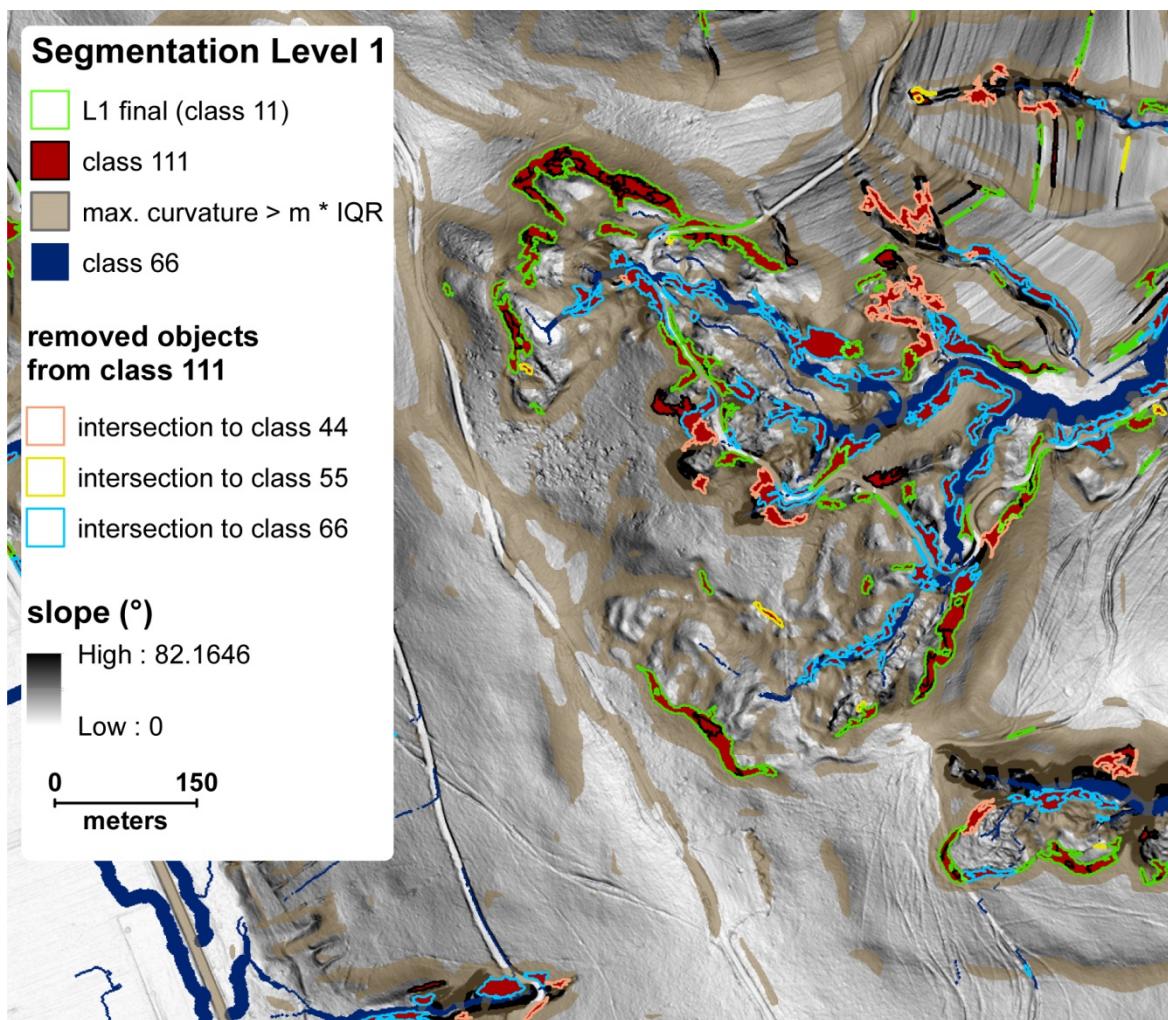


Figure 5-2: Segmentation level 1 (OWN REPRESENTATION).

In the last section, the *L1\_final* result was processed using shape metrics and by calculating the mean flow direction. Again, thresholds were calculated using k-means thresholding and are tabularized in Table 5-2.

Table 5-2: Level 1 – k-means thresholds of shape metrics (OWN REPRESENTATION).

Level 1: k-means thresholds				
<i>n</i>	<b>convexity</b>	<b>compactness</b>	<b>shape index</b>	<b>length-width-ratio</b>
1	0.4667959*	0.07409421*	2.082387	6.612209
2	0.4939739* <sup>3</sup>	0.14756019* <sup>3</sup>	2.902504	23.23219
3	0.6868825	0.2315563	4.057335	47.635474
4		0.34751494	4.7* <sup>4</sup>	86.942898
5			6.112338	145.223219*
6				220.831783
7				336.999609
8				581.615315

Level 1: k-means thresholds				
<i>n</i>	<i>convexity</i>	<i>compactness</i>	<i>shape index</i>	<i>length-width-ratio</i>
9				1319.96005

All thresholds calculated for whole dataset. \* Selected threshold for further analysis. † Refined after visual inspection. \* Refined threshold for segmentation level 3. x Also used in segmentation level x.

According to the characteristics of a scarp (s. Table 4-1), and after empirical examination of the thresholds by visual inspection of the data, the following expression was used to refine the objects of a potential landslide scarp (*class 11*):

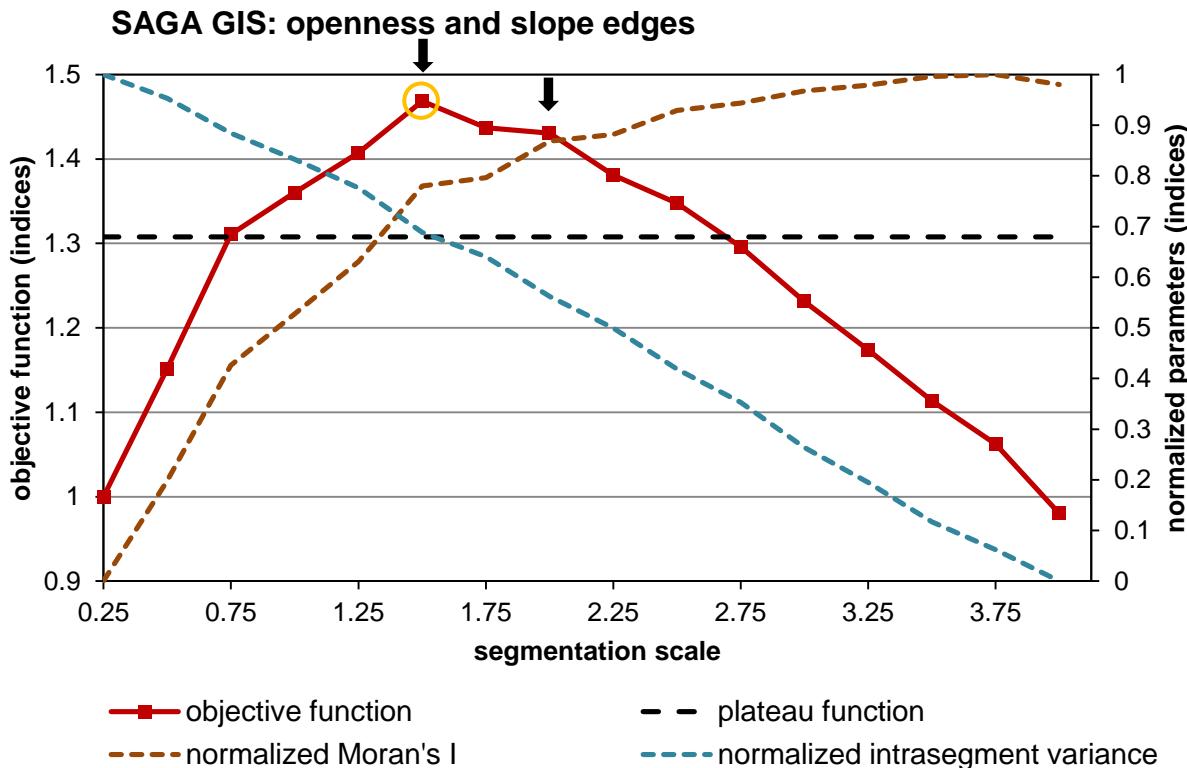
```
class 11 = main direction ⊥ flow direction == 1
  & length width ratio < 145.223219
  & shape index < 4.7
  & area < 5,000
  | (main direction ⊥ flow direction == 0
    & convexity < 0.4667959
    & compactness ≥ 0.07409421)
```

The maximum area of 5,000 m<sup>2</sup> was empirically set. It was assumed that no landslide scarp in the study area could have a greater area. Besides, due to the semi-circularity of a scarp, the perpendicular threshold between main and flow direction was set to the range of 63° to 117° (30 % derivation of 90°). However, for the case that for some scarps the condition of perpendicular was still not fulfilled, the convexity threshold was added. By this query, the potential scarps could be reduced to 3,220 objects.

Finally, *L1\_final* was rasterized (*L1\_final.sdat*) and buffered with a 5 m buffer distance (*L1\_final\_buf.sdat*). While the rasterization served as boundary input in the next segmentation level, the buffer should restrict the area of possible landslide flanks. To retrieve the possible landslide scarps as well as their attributes in the further analysis, one point on every object's surface of *L1\_final* was created (*L1\_final\_pts.shp*).

### 5.1.2 Segmentation Level 2

Segmentation level 2 had the objective to detect potential landslide flanks. However, the segmentation was also suitable to detect potential landslide crowns using the maximum curvature according to TAROLLI et al. (2012:71) (s. 4.1.2.1).



Black arrows: optimal scale parameters. Yellow circle: selected scale.  
Segmentation scale represents the Level of Generalisation in the module Fast Representativeness.

Figure 5-3: Level 2 – Objective function (own REPRESENTATION).

The second segmentation level was also based on a fine-scale segmentation. Therefore, the segmentation scales ranging from 0.25 to 4 were investigated by the objective function (Figure 5-3, Table B-2 in the Appendix B). As segmentation input the openness was combined with slope edges, and the statistics were calculated based on the openness values.

As the graph in Figure 5-1 shows, all segmentation scales ranging from 0.75 to 2.5 had higher OF values than the plateau function value (1.3077). Here, two peaks are visible representing the local optimal segmentation scale: one at scale 1.5 with the OF value 1.469 and one at scale 2.0 with the OF value 1.430. For further processing, the segmentation scale with the highest OF value (scale 1.5) was chosen.

Due to the possibility of hierarchical segmentation in GRASS GIS, the second segmentation resulted from a combination of a SAGA and GRASS GIS segmentation including the result of the first segmentation level (*L1\_final*).

At first, in SAGA GIS a segmentation was computed using the implemented seeded region growing algorithm. The seeds resulted from the module *Fast Representativeness* with the optimal scale parameter of the OF above as *Level of Generalisation*. Subsequently, the SAGA GIS segmentation result was used as seed input for the GRASS GIS segmentation. However, to maintain *L1\_final* in the GRASS GIS segmentation, *L1\_final* was “burned” into

the SAGA GIS segmentation result. Moreover, *L1\_final* served as boundary grid. It was assumed that the SAGA GIS segmentation result (based on the optimal scale) would retain optimal in the GRASS GIS segmentation by using a very small threshold. Therefore, the threshold in *i.segment* was set to 0.0001, and no other thresholds were tested by the OF.

The segmentation of L2 resulted in 292,404 different objects (*L2\_seg.shp*). For the reduction of objects an intersection with *L1\_final\_buf* was performed, and resulted in 25,269 objects (*L2\_seg\_sel.shp*). Based on this intersection the thresholds for different land surface parameters and shape metrics were computed by using k-means thresholding, and are tabularized in Table 5-3.

**Table 5-3: Level 2 – k-means thresholds of land surface parameters and shape metrics (own representation).**

<b>Level 2: k-means thresholds</b>						
<i>n</i>	<i>slope</i>	<i>openness</i>	<i>maximum curvature</i>	<i>shape index</i>	<i>length-width-ratio</i>	<i>maximum distance-area-ratio</i>
1	7.072568	-1.8480002*	0.00063656	1.925149	3.027028	0.06748803
2	13.407496*	-0.1382933	0.00534929	2.621215	7.448578	0.10131988*
3	19.588355*	1.0989332	0.01154292*	3.453631*	13.321072	0.13805622
4	26.283062		0.02147743	4.858446	20.964592	0.18113174
5	34.615099				30.692215	0.22809245
6					42.75948	0.2797809
7					58.601485*	0.33752436
8					78.964692	0.40357238
9					103.913564	0.48833247
10					135.413909	0.59517102
11					176.960206	0.74288104
12					232.95437	1.2335866
13					315.449842	
14					420.016491	
15					578.089453	
16					865.326689	
17					1246.27978	

\* Selected threshold for further analysis.

According to the characteristics of a flank (s. Table 4-1) and the further assumptions made (s. 4.1.2.2), suitable thresholds based on the k-means clustering results (Table 5-3) were empirically examined by visual inspection of the data.

As first step, the differences in the main direction of each neighbor to a scarp object were calculated using the *RelationalClassFunction* with the specific angle-argument. Based on that calculation, those objects were attributed as potential flank ( $== 1$ ) whose main direction had an angle perpendicular ( $90^\circ \pm 30\%$ ) to the direction of its scarp. However, to take scarps with a very low convexity value into account, the threshold was extended to  $90^\circ \pm 50\%$  for objects with a convexity value smaller than 0.4667959 (s. Table 5-2). Furthermore, two bounding boxes were created from each scarp. They had the objective to attribute neighbor objects lying in flow or in inverse flow direction. The extent of each bounding box has a high impact of the selected and thus, attributed objects. However, the settings in Table A-1 (Appendix A) were found suitable in this investigation (Figure 5-4).

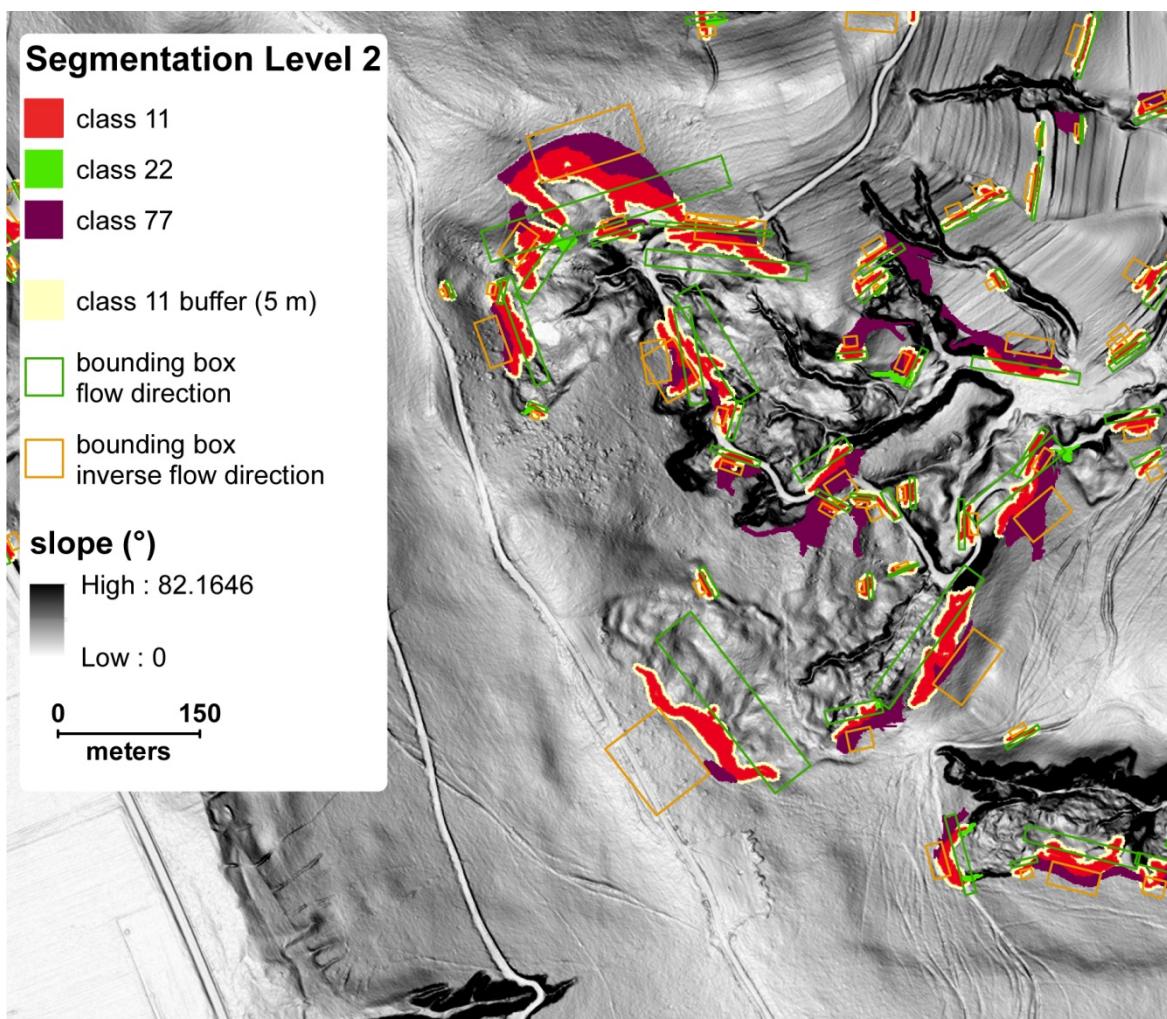


Figure 5-4: Segmentation level 2 (own representation).

In the next step, potential flanks (class 22) were classified based on the following query:

```
class 22 = flank == 1
& neighbor in flow direction == 1
& slope ≥ 13.407496
& openness ≥ -1.8480002
```

---

```

& distance area ratio ≥ 0.10131988
& shape index < 3.453631
& length width ratio < 58.601485
& maximum curvature < 0.0115429194
& area ≥ 25

```

The minimum area of 25 m<sup>2</sup> was adapted from the step in L1. After the selection, the *class* 22 objects were further refined by filtering those flanks whose area was greater than the weighted-mean area of the belonging scarps. Thus, 359 objects were attributed as potential landslide flank.

Besides the detection of flanks, the resulted objects of this segmentation level also showed great potential to identify possible landslide crowns (*class* 77). Here, it was assumed that a crown is located in inverse flow direction of a scarp and its main direction should be parallel to the main direction of the scarp. In addition, according to TAROLLI (2014) a crown shows high values in maximum curvature. Therefore, the following expression selected 1,168 objects describing potential landslide crowns:

```

class 77 = slope < 19.588355
& neighbor in inverse flow direction == 1
& maximum curvature ≥ 0.0115429194
& area ≥ 25

```

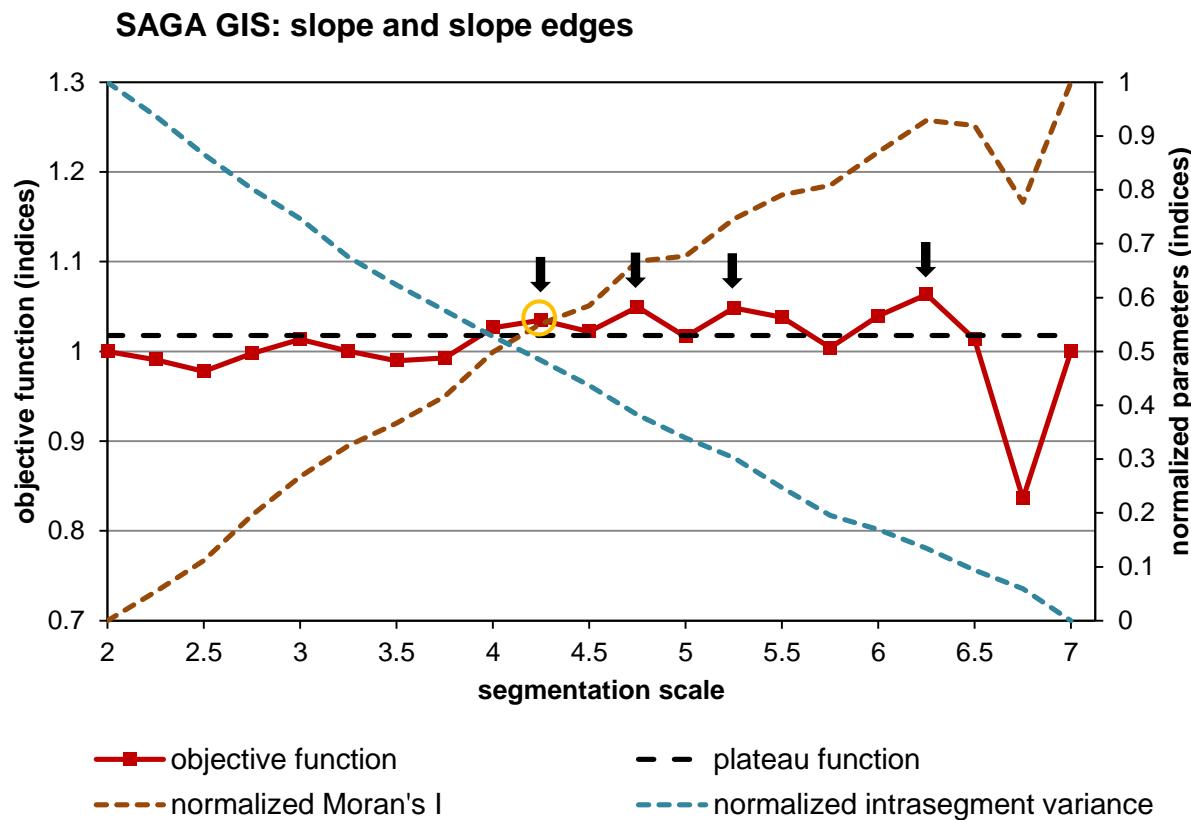
The minimum area of 25 m<sup>2</sup> was set according to the steps done in L1. The assumption of the similarity of the main directions between scarp and crown (||) could not be checked in this segmentation level.

Due to the fine segmentation, it was possible that multiple objects presented one flank or crown. Thus, by the function *NeighborGrowing* neighbors of the same class were merged. This resulted in the final output for flanks (*L2\_flank.shp*) consisting of 331 objects, and for crowns (*L2\_crown.shp*) consisting of 794 objects.

In the last step of L2, *L2\_flank* and *L2\_crown* were rasterized (*L2\_flank.sdat*, *L2\_crown.sdat*). To receive a file containing all relevant possible landslide parts created up to here, *L1\_final*, *L2\_flank* and *L2\_crown* were added together and every object was kept with a unique value (*L2\_final.sdat*). Then, *L2\_final* was buffered with a 25 m buffer distance (*L2\_final\_buf.sdat*). While the *L2\_final* served as boundary input in the last segmentation level, the buffered output should restrict the area of possible landslide bodies. To retain the attributes and positions of the possible landslide crowns and flanks in the further analysis, one point on every object's surface of *L2\_flank* and *L2\_crown*, respectively, was created (*L2\_flank\_pts.shp*, *L2\_crown\_pts.shp*).

### 5.1.3 Segmentation Level 3

Segmentation level 3 was the last segmentation step and had the objective to detect the landslide body. In addition, the landslide scarps were further refined and finally the different landslide parts were grown together into one single landslide element.



Black arrows: optimal scale parameters. Yellow circle: selected scale.  
 Segmentation scale represents the Level of Generalisation in the module Fast Representativeness.  
**Figure 5-5: Level 2 – Objective function (own representation).**

In L3 coarser segmentation scales than in L1 or L2 were investigated by the objective function: scales ranging from 2 to 7 (Figure 5-5, Table B-3 in the Appendix B). Likewise L2, the result of the third segmentation was a combination of a SAGA and GRASS GIS segmentation including the results of L1 and L2 (*L2\_final*). The segmentation was performed based on the combination of slope and slope edge, and the statistics were calculated based on the slope values.

However, in the graph there is no clear domain with optimal parameters. Instead, the curve goes up and down creating multiple peaks representing local optimal segmentation scales. The peaks are located at scale 4.25 (OF value 1.035), 4.75 (OF value 1.049), 5.25 (OF value 1.049), and 6.25 (OF value 1.063). After visual examination of the optimal scales, the scale 4.25 with a OF value of 1.035 was selected for the further analysis. Although the other optimal scales had higher OF values, their resulted objects were already larger than some small entire landslides.

In the first step of L3, the SAGA GIS segmentation (*Fast Representativeness and Seed Generation* with *Level of Generalisation* of scale 4.25) was used as seed input for the GRASS GIS segmentation. Likewise in L2, the results of the previous levels (*L2\_final*) were “burned” into the segmentations to keep their specific shape and location also in L3. In addition, *L2\_final* served as boundary grid in the GRASS GIS segmentation. In order to maintain the SAGA GIS segmentation with the optimal scale parameter also after computing the GRASS GIS segmentation, the threshold in *i.segment* was set to the very low value of 0.0001. Therefore, no other thresholds were tested by the OF.

The segmentation in L3 created 65,229 different objects (*L3\_seg.shp*). By the intersection with *L2\_final\_buf*, the amount of objects in *L3\_seg* could be reduced to 23,170 representing locations for potential landslide bodies (*L3\_seg\_sel.shp*). In the following, based on the objects in *L3\_seg\_sel* the thresholds for different land surface parameters, shape metrics and textural features were computed by using k-means thresholding. In addition, thresholds were calculated for neighbor statistics of the potential scarps (class 11) using the result of the *ClassNeighborFunction* on slope, sky-view factor, entropy<sub>Flow</sub> and variance<sub>Flow</sub>.

To determine neighbors and objects located in flow direction of a scarp, a bounding box scaled and rotated in flow direction for each scarp was created. However, to take the convexity of the scarps into account, the bounding box resulted from a combination of two bounding boxes with different settings in dependence on a convexity threshold (< 0.4667959, s. Table 5-2) (settings in Table A-1, Appendix A).

The thresholds of the land surface parameters are tabularized in Table 5-4, for the results of the *ClassNeighborFunction* on land surface parameters in Table 5-5, and in Table 5-6 the thresholds of shape metrics, of textural features and of the results of the *ClassNeighborFunction* on textural features are shown.

**Table 5-4: Level 3 – k-means thresholds of land surface parameters (own representation).**

<b>Level 3: k-means thresholds</b>						
<i>n</i>	<i>normalized height</i>	<i>openness</i>	<i>surface roughness</i> *	<i>sky-view factor</i>	<i>slope</i>	<i>SD of slope</i>
1	0.08239538*	-3.2177916*	0.00100278	0.7101694	4.60608	1.741864
2	0.19024493*	-2.183391*	0.00229322	0.7865066	8.821035*	3.311322
3	0.29894804	-1.1489897	0.00372515*	0.8407246	13.321149	4.919841*
4	0.41329221	0.1280833	0.00536189	0.893389	18.094487	6.858663
5	0.5395143	1.5434541	0.00720045	0.9435241*	23.17667*	9.728644
6	0.68107097		0.00914532		26.54378*	
7	0.84494128		0.01109139*		27.504224*	
8			0.01311272		32.481872	

Level 3: k-means thresholds						
<i>n</i>	<i>normalized height</i>	<i>openness</i>	<i>surface roughness</i> *	<i>sky-view factor</i>	<i>slope</i>	<i>SD of slope</i>
9			0.01520632		39.046244	
10			0.01731803			
11			0.01962795			
12			0.02210885			
13			0.03304413			
14			0.04487619			
15			0.05902739			
16			0.0763508			
17			0.10282954			
18			0.1485526			

\* Selected threshold for further analysis. ♦ Refined after visual inspection. ^ Extended thresholds due to screwed data. SD: standard deviation.

Table 5-5: Level 3 – k-means thresholds of *ClassNeighborFunction* on land surface parameters (own REPRESENTATION).

Level 3: k-mean thresholds							
<i>n</i>	<i>slope mw</i>	<i>slope mwfM</i>	<i>slope mwfA</i>	<i>slope mdfM</i>	<i>slope mdfA</i>	<i>sky-view factor mwfM</i>	<i>sky-view factor mwfA</i>
1	7.496668	8.157438	6.75608*♦	0*♦	0*♦	0.8027135	0.809069
2	13.139954*	15.771015	9.791803*	6.75608*	7.839448	0.8674388	0.8742489
3	19.317878	24.904971	20.498408	19.09039	19.541741	0.9239724*	0.9274646*
4	26.728982						

\* Selected threshold for further analysis. ♦ Refined after visual inspection. mdfM: weighted mean-difference to neighbors in flow direction. mdfA: weighted mean-difference to all objects in flow direction. mw: weighted mean. mwfA: weighted mean to neighbors in flow direction. mwfM: weighted mean to all objects in flow direction.

Table 5-6: Level 3 – k-means thresholds of textural features and shape metrics (own REPRESENTATION).

Level 3: k-means thresholds						
<i>n</i>	<i>entropy flow</i>	<i>entropy flow mdfM</i>	<i>variance flow mdfA</i> ^	<i>shape index</i>	<i>length-width-ratio</i>	<i>interior edge ratio</i>
1	2.735676	0*♦	-19.042292	1.956049	2.523199	0.2220594
2	3.545	0.06250103	-12.7461	2.678501	5.353208	0.4360765
3	4.042022*	0.78817889	-2.210978	3.453671*	8.880954	0.7110833*
4	4.506518	1.54810914	6.46147	3.8*♦	13.116816	1.0533773
5	4.923196*		12.967614	4.472944*	18.194977*	1.4782892

Level 3: k-means thresholds						
<i>n</i>	<i>entropy flow</i>	<i>entropy flow mdfM</i>	<i>variance flow mdfA</i> *	<i>shape index</i>	<i>length-width-ratio</i>	<i>interior edge ratio</i>
6			30.753428	6.11469	24.013895*	3.0752165
7			47.220238		30.95963	
8			60*♦		38.942666*	
9			80*♦		48.166087	
10					58.963847*	
11					73.031801	
12					88.978868	
13					109.061696	
14					134.844211	
15					168.804848	
16					214.733364	
17					284.377258	
18					373.194567	
19					485.383643	
20					716.831079	
21					1173.67779	
22					4669.63724	

\* Selected threshold for further analysis. ♦ Refined after visual inspection. ^ Extended thresholds due to screwed data. mdfM: weighted mean-difference to neighbors in flow direction. mdfA: weighted mean-difference to all objects in flow direction.

Before selecting landslide body parts, the potential landslide scarps of L2 (class 11) were further refined with the objective to remove possible false-positive candidates. Especially the information on their neighborhood was suitable for the selection. The thresholds for the following queries were chosen empirically after visual examination of the data.

At the beginning, three more classes were created (Figure 5-6):

- class 88: normalized height (< 0.08239538),
- class 89: normalized height (< 0.19024493),
- class 99: length-width-ratio ( $\geq$  38.942666).

Class 88 and 89 represented objects at the bottom of a slope relative to their local environment. Class 99 consisted of straight, narrow and long objects. Subsequently, for these classes a class-to-class relationship was performed to class 11 by the *RelationalClassFunction*. This operation had the objective to receive the relative border of class 11 to these classes in order to unselect false-positive candidates such as objects along agricultural fields, rivers or streets.

In the next step, all class 11 objects were attributes as class 10. Then, the following selection removed objects from class 10 and re-attributed them as class 11:

```

# remove objects whose slope is less than their neighbors
class 11 = slopemdfM < 0 | slopemdfA < 0
# remove objects with small entropy values
| (slopemdfM < 6.756081 & convexity ≥ 0.4939739 & slopemwfM ≥ 0
& entropyflow < 4.923196)
# remove objects in high sky view factor neirhborhood
| (sky view factormwfM ≥ 0.9239724 & convexity ≥ 0.4939739
| sky view factormwfA ≥ 0.9274646 & slopemwfA < 6.756081)
# remove of earth banks and roadside ditches
| (varianceflow mdfA ≥ 60 & slopemwfA ≤ 9.791803)
| (interior edge ratio ≥ 0.7110833 & compactness < 0.14756019
& varianceflow mdfA ≥ 80)
| (area ≥ 850 & entropyflow mdfM < 0 & slopemw ≥ 13.139954)
| (length width ratio ≥ 18.194977 & entropyflow mdfM < 0)
| intersection(length width ratio( ≥ 58.963847) with class 10)
| (rel.borderto class 90 ≥ 10 | rel.borderto class 89 ≥ 75 )
| (rel.borderto class 88 ≥ 40 | rel.borderto class 89 > 0 )
| (area ≥ 350 & rel.borderto class 88 > 0 & rel.borderto class 89 ≥ 25 )

```

With  $m$  for mean,  $d$  for difference,  $w$  for weighted,  $f$  for in flow direction,  $M$  for the matching with direct neighbors, and  $A$  for all neighbors. Therefore, for example  $mdfM$  stands for “weighted mean-difference to neighbors in flow direction”, or  $mwfA$  for “weighted mean to all objects in flow direction”. The thresholds for area or relative border were empirically set. By this unselection 1,736 objects of class 10 could be assigned to class 11.

Next, based on the objects of class 10, statistics for their neighbors in flow direction were calculated by using the *RelationalClassFunction* on slope, DTM, area, shape index and length-width-ratio. Again, thresholds were computed using the *kMeansThresholds* function (Table 5-7).

**Table 5-7: Level 3 – k-means thresholds of RelationalClassFunction for class 10 (OWN REPRESENTATION).**

Level 3: k-means thresholds						
<i>n</i>	<i>DTM</i> <i>md2Cl</i>	<i>slope</i> <i>md2Cl</i>	<i>slope</i> <i>mw2Cl</i>	<i>area</i> <i>r2Cl</i>	<i>length-</i> <i>width-ratio</i> <i>mw2cl</i>	<i>shape index</i> <i>mw2Cl</i>
1	-15.476275	-21.073871	14.04851	0.1773351	5.145388	2.601358
2	-10.76794	-13.30907	26.56008*	0.7331048*	12.971049*	3.70509*
3	-7.554506	-5.491192	26.54378*	0.1*	24.186127	3.8*

Level 3: k-means thresholds						
<i>n</i>	<i>DTM md2CI</i>	<i>slope md2CI</i>	<i>slope mw2CI</i>	<i>area r2CI</i>	<i>length- width-ratio mw2cl</i>	<i>shape index mw2CI</i>
4	-4.931933	0*♦		1*♦	30.959630*♦	4.47294*♦
5	-2.826722	1.409017*		1.3596988	43.66481	5.228935*
6	-0.937815*	4.336116		2.0977633	82.280344	
7	0*♦	7.758105		2.9970024	148.080811	
8	0.916749	12.848418		3.9494035	463.008127	
9	2.840925*			5.1864926*		
10	4.913497			6.5547397*		
11	7.415527			8.0506983		
12	10.936923			9.4997718		
13	17.214348			11.1052252		
				14.473987		
				24.306542*		
				37.315021*		
				54.347445		
				78.131136		
				115.310797		
				166.458431		
				253.175149		
				392.809488		

\* Selected threshold for further analysis. ♦ Refined after visual inspection. md2CI: weighted-mean difference to class. mw2CI: weighted-mean to class. r2CI: ratio of object to its class objects.

The following selection was performed based on the characteristics of a landslide body (s. Table 4-1) and on the investigated thresholds of the k-means clustering results (class 30):

```

# keep objects located in flow direction of a scarp
class 30 = object in flow direction == 1
# keep objects with high entropy and low sky view factor values
& ((entropyflow ≥ 4.042022 & sky view factor < 0.9435241)
# keep objects with a smaller slope and elevation than scarp, and high surface roughness
& (slopemd2cl < 1.409017 & DTMmd2cl < 2.840925
    & surface roughness ≥ 0.003725149)
# remove neighbor objects not in flow direction: steep, valley, big area, long and narrow shape
& !(objectNB in flow direction == 0 & (area ≥ 2000
    | length width ratio ≥ 24.013895 | shape index ≥ 4.47294
    | slope ≥ 27.504224 | openness < -3.2177916 | normalized height < 0.08239538))
# remove extremely steep neighbor objects of scarp

```

```

& !(slopemw2Cl ≥ 26.54378 & (area ≥ 150 & rel.borderto class 10 < 10
| slopemd2Cl > 0))
# remove extremely large neighbor objects of scarp
& !(arear2Cl ≥ 6.5547397 & (slope ≥ 27.504224 | slope < 8.821035
| surface roughness < 0.005361888 | normalized height < 0.19024493
& rel.borderto class 10 < 5.977698 | DTMmd2Cl > 0 | slopemd2Cl > 0)
| (arear2Cl ≥ 37.315021 | areamd2Cl ≥ 5000))

# remove objects smaller than their scarp
& !(areamd2Cl < 0 & arear2Cl < 0.7331048 & rel.borderto class 10 < 15
| arear2Cl < 0.1 & rel.borderto class 10 < 50
| arear2Cl < 1 & (slopemd2Cl > 0 | DTMmd2Cl > 0))

# removal of long and narrow objects: earth banks and roadside ditches
& !(shape indexmw2Cl ≥ 3.705090 & (rel.borderto class 10 < 5
| normalized height < 0.0823953 | slope < 8.821035 | slope ≥ 26.54378
& DTMmd2Cl < -0.937815) | shape indexmw2Cl ≥ 5.228935
| (shape indexmw2Cl ≥ 4.47294 & arear2Cl ≥ 5.1864926)
& !(length width ratiomw2Cl ≥ 12.971049 & slope < 8.821035
& (rel.borderto class 10 < 29 | DTMmd2Cl > 0 | slopemd2Cl > 0)
| length width ratiomw2Cl ≥ 30.959630)

# removal of objects located totally inside of a scarp
& !(rel.borderclass 10 == 100))

```

With *mw2Cl* for “weighted-mean to class”, *md2Cl* for “weighted-mean difference to class”, *r2cl* for “ration between class and object”, and *NB* for neighbor of class. The area or relative border thresholds in this query were empirically set without the use of k-means thresholding. By this selection, from the 6,503 objects located in flow direction of the scarps 1,792 objects were attributed as potential landslide body (*class 30*).

For this thesis, a landslide was detected when it consisted at least of a scarp and a body. Hence, the resulting objects of the intersection between class 10 and class 30 represented the potential final scarp objects (class 1, here 705 objects). However, due to the bounding box extent, there was the possibility that objects upslope of a scarp were attributed as “located in flow direction”, even if they were not. Therefore, likewise the bounding box in flow direction, a bounding box in inverse flow direction was computed, but with smaller scale factors (settings in Table A-1 in the Appendix A). Besides, by the *RelationalClassFunction* the relative border from class 1 to class 30 was calculated. With that information, class 10 and 30 were further refined by unselecting objects with the following expression:

```

class 1 = class == 1
& (rel.borderto class 30 == 100
| rel.borderto class 30 == 0
| rel.borderto class 30 ≥ 90 & convexity ≥ 0.4939739

```

|  $\text{rel.border}_{\text{to class 30}} < 14 \text{ & shape index} \geq 3.8$ )

The thresholds for the relative border were empirically set. However, in their selection it was considered, that ideally the relative border of a scarp to its body should not be more than 51% of the whole scarp border.

```
NULL = class == 30
& object in inverse flow direction == 1
& (arear2Cl ≥ 24.306542
| DTMmd2Cl > 0 & rel.borderto class 10 < 30)
```

By that expression 56 large objects located in upslope direction of the scarp as well as those with a higher elevation were removed from class 30 (1,736 objects).

Similar to the previous step, the refined classes 1 and 30 were intersected again. The result represented the final landslide scarps consisting of 640 objects (class 1).

The next step had the objective to finally detect the final landslide bodies. At first, the bounding box in flow direction as well as the objects located in flow direction was adjusted in relation to the remaining scarps. Then, the *RelationalClassFunction* was performed on the merged classes 1 and 30 (class 130) to the remaining objects in order to receive information about their relative border. Finally, the landslide bodies were selected by following expression:

```
# select objects with a border to class 130
class 3 = rel.borderto class 130 > 0
# select objects with a high relativ border, small shape index, and certain slope
& ((object in flow direction == 1 & rel.borderto class 130 ≥ 60)
| (rel.borderto class 130 != rel.borderto class 130
& object in inverse flow direction == 0 & shape index < 4.472944
& rel.borderto class 130 ≥ 51 & objectNB in flow direction == 1
& slope ≥ 8.821035)

# select objects with a hummocky topography: surface roughness, slope SD
| (rel.borderto class 130 ≥ 20 & 8.821035 ≤ slope < 23.17667
& ((surface roughness ≥ 0.01109138 & shape index ≤ 3.453671
& object in inverse flow direction == 0)
| (slope SD ≥ 4.919841 & object in inverse flow direction == 0)))

# select large objects with a small shape index
| (rel.borderto class 130 ≥ 20 & area ≥ 4300
& shape index ≤ 3.453671 & objectNB in flow direction == 1))
```

With *SD* for standard deviation and *NB* for neighbor of class. Area and relative border thresholds were empirically set after visual investigation of the data. By this query 507 objects were selected. Subsequently, these objects were merged to class 30, and the resulting 2,243 objects represented the final selection of landslide bodies (class 3) (Figure 5-6).

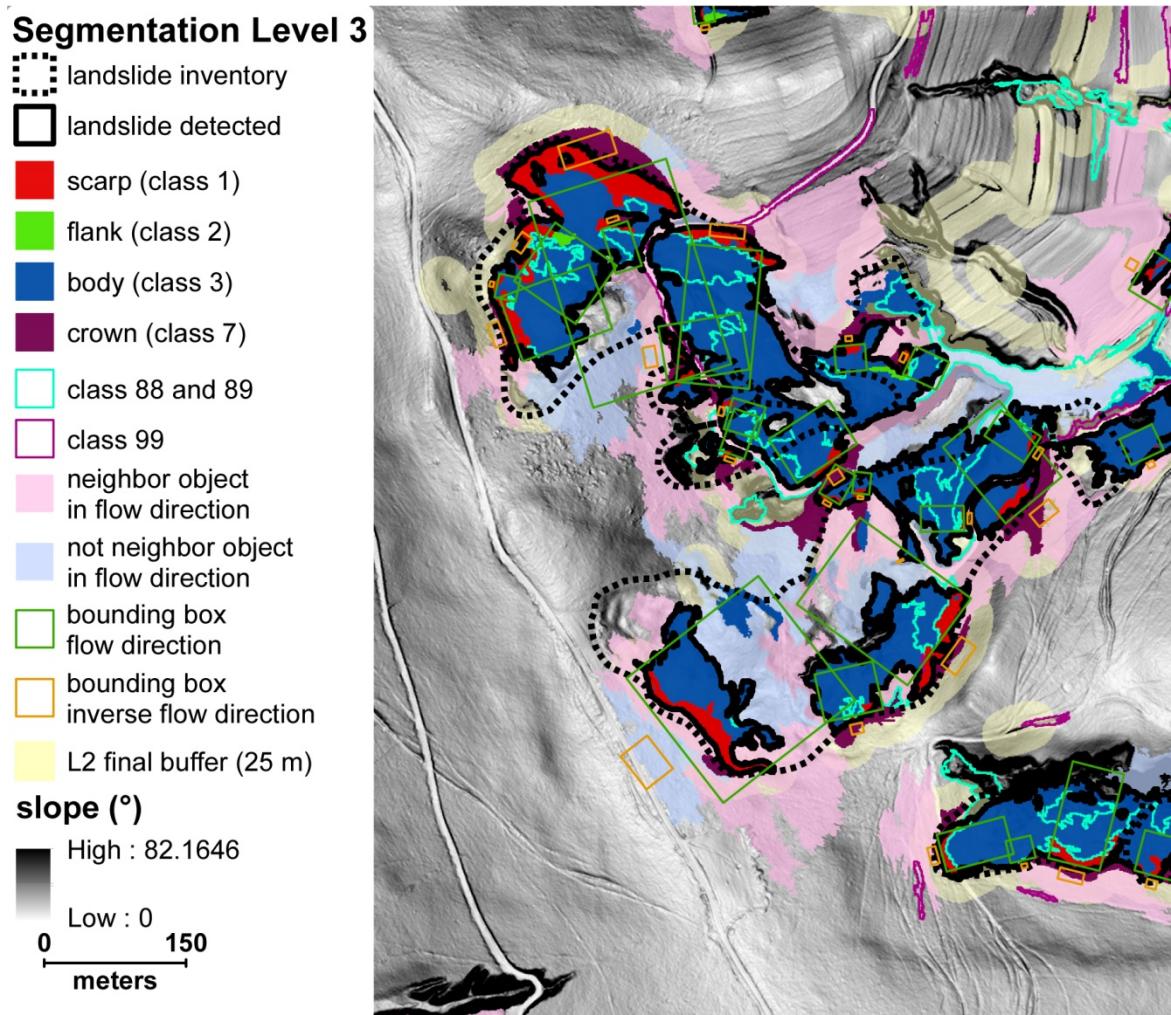


Figure 5-6: Segmentation level 3 (own representation).

In the last step, flanks (class 22) and crowns (class 77) were selected by their intersection to class 1: class 2 (flanks) with 116 objects and class 7 (crowns) with 304 objects. Finally, class 1, 2, and 3 were grown together into one single landslide object by using the *NeighborGrowing* function. By that, 398 different landslides were detected.

## 5.2 Accuracy Assessment and Comparison

The area of the 398 detected landslides (DT) ranged from 0.008 (inventory: 0.04, s. Table 3-2) to 3.56 ha (inventory: 17.22) (mean: 0.4 ha, inventory: 1.23) and in total 158.27 ha (inventory: 178.86). Only 160 of the DL were classified as correct (43.36%) (Table 5-8). However, the overall area of the correctly classified landslides (TL, true landslides) was 101.62 ha and thus 68.89% of the DL. Around 79.31 ha were correctly detected inside the inventory. That was 44.34% of the entire area in the inventory, but 78.05% related to TL. Outside the inventory, circa 22.31 ha were additionally detected as TL (21.95%). In relation to the area of the inventory, that was a plus of 12.47%.

Correctly classified scarps were located in 91 of 148 inventory landslides resulting in a true positive rate (TP) of 61.47%. Considering the TL, 97 landslides of the inventory were detect-

ed (TP: 65.54%). The difference of six landslides can be attributed to scarps whose body grew into more than one inventory landslide. However, referring to the assessment approach of VAN DEN EECKHAUT et al. (2012) using the 50% threshold for scarp and body (s. 4.1.4, Table 5-9) only 52 landslides were correctly detected (TP: 35.14%). Even by decreasing the threshold to 15%, the TP remained still low with 58.78%. The positional mismatch between landslide classification and inventory was 78.70% considering all landslides, and 74.29% considering only the correctly detected landslides and the inventory (Table 5-10).

**Table 5-8: Accuracy assessment of landslide classification (OWN REPRESENTATION).**

		landslide classification		landslides	
		scarps (n)	flanks (n)	n	A (ha)
... landslide classification	<b>total</b>	640 (604)	116 (115)	398 (369)	158.57 (147.52)
	<b>uncertain*</b>	36	1	29	10.74
	<b>correct classified</b>	308	54	160	101.62
	<b>TP (%)</b>	<b>50.99</b>	<b>46.96</b>	<b>43.36</b>	<b>68.89</b>
... inside inventory	<b>total</b>	199	32	98	79.31
	<b>correct classified</b>	198	27	98	79.31
	<b>TP (%)</b>	<b>99.50</b>	<b>84.38</b>	<b>100</b>	<b>100</b>
	<b>... outside inventory</b>				
... outside inventory	<b>total</b>	401	83	271	68.21
	<b>correct classified</b>	110	27	62	22.31
	<b>TP (%)</b>	<b>27.43</b>	<b>32.53</b>	<b>22.88</b>	<b>32.71</b>

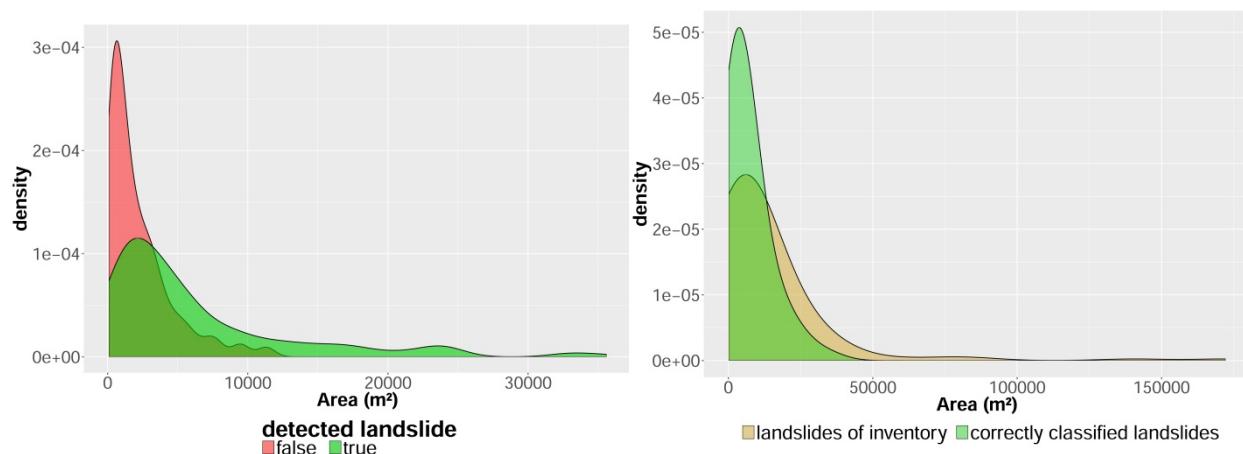
\* Uncertain about true or false candidate, and therefore excluded in accuracy assessment. n: number.

**Table 5-9: Comparison between landslide inventory and classification (OWN REPRESENTATION).**

landslide inventory (n = 148)		
	N	TP (%)
<b>scarp</b>	91	61.47
<b>scarp + body*</b>	97	65.54
<b>scarp + body* <math>\geq</math> 50%</b>	52	35.14
<b>scarp + body* <math>\geq</math> 30%</b>	71	47.97
<b>scarp + body* <math>\geq</math> 15%</b>	87	58.78

N: number of landslides in classification that correspond to landslide in inventory. TP: true positive rate or percentage of correctly identified landslides in inventory by classification. \* Flank is included in scarp + body.

The area distribution in the classification indicated that particularly small landslides were overrated and wrongly classified (Figure 5-7, left). From 209 wrongly classified landslides, 141 were smaller than 2,500 m<sup>2</sup> (67.46%). Only 59 landslides were correctly identified (36.88%). The area of the largest wrongly detected landslide was 11,574 m<sup>2</sup>. However, around 17.5% of the correctly detected landslides had greater areas (number of 28). By comparing the area distribution of correctly classified landslides with the inventory landslides, it is shown that the classified landslides were generally smaller (Figure 5-7, right).



Left: area distribution in dependence of true/false landslides in classification.

Right: area distribution of inventory and correctly classified landslides.

**Figure 5-7: Distribution of area in classification and inventory (OWN REPRESENTATION).**

Looking at the CLC2012 classification of the landslide areas (Table 5-10), around 4.27% of the correctly detected landslides were located on agricultural areas (4.34 ha, inventory: 5.97%), 95.71% on forest and semi natural areas (97.26 ha, inventory: 90.25%), and 0.02% on areas labeled as artificial surfaces (0.02 ha, inventory: 3.78%). The distribution of percentage shares looks therefore similar. However, considering the absolute values, around 6.34 ha less were classified as landslides on agricultural areas, 6.74 ha less on areas labeled as artificial surfaces and 64.16 ha less on forest and semi natural areas than in the

inventory. Besides, correctly detected landslides that were missed in the inventory were mainly located on forest and semi natural areas (88.83%).

**Table 5-10: Landslide classification statistics for the study area and inventory (own REPRESENTATION).**

parameter	agricultural areas (ha, %)	forest and semi natural areas (ha, %)	artificial surfaces (ha, %)	$A_{E \cup O}$ (ha)	$A_{E \cap O}$ (ha)	PM (%)
<b>landslide classification</b>	13.60 (8.6%)	143.25 (90.51%)	1.41 (0.89%)	59.20	277.93	78.70
<b>correctly detected landslides</b>	4.34 (4.27%)	97.26 (95.71%)	0.02 (0.02%)	57.36	223.13	74.29
<b>correct landslides missed in inventory</b>	2.47 (11.06%)	19.82 (88.83%)	0.02 (0.11%)			

PM: position mismatch.  $A_{E \cup O}$ : united area of landslide detection and inventory.  $A_{E \cap O}$ : intersected area of landslide detection and inventory.

## 6 Discussion

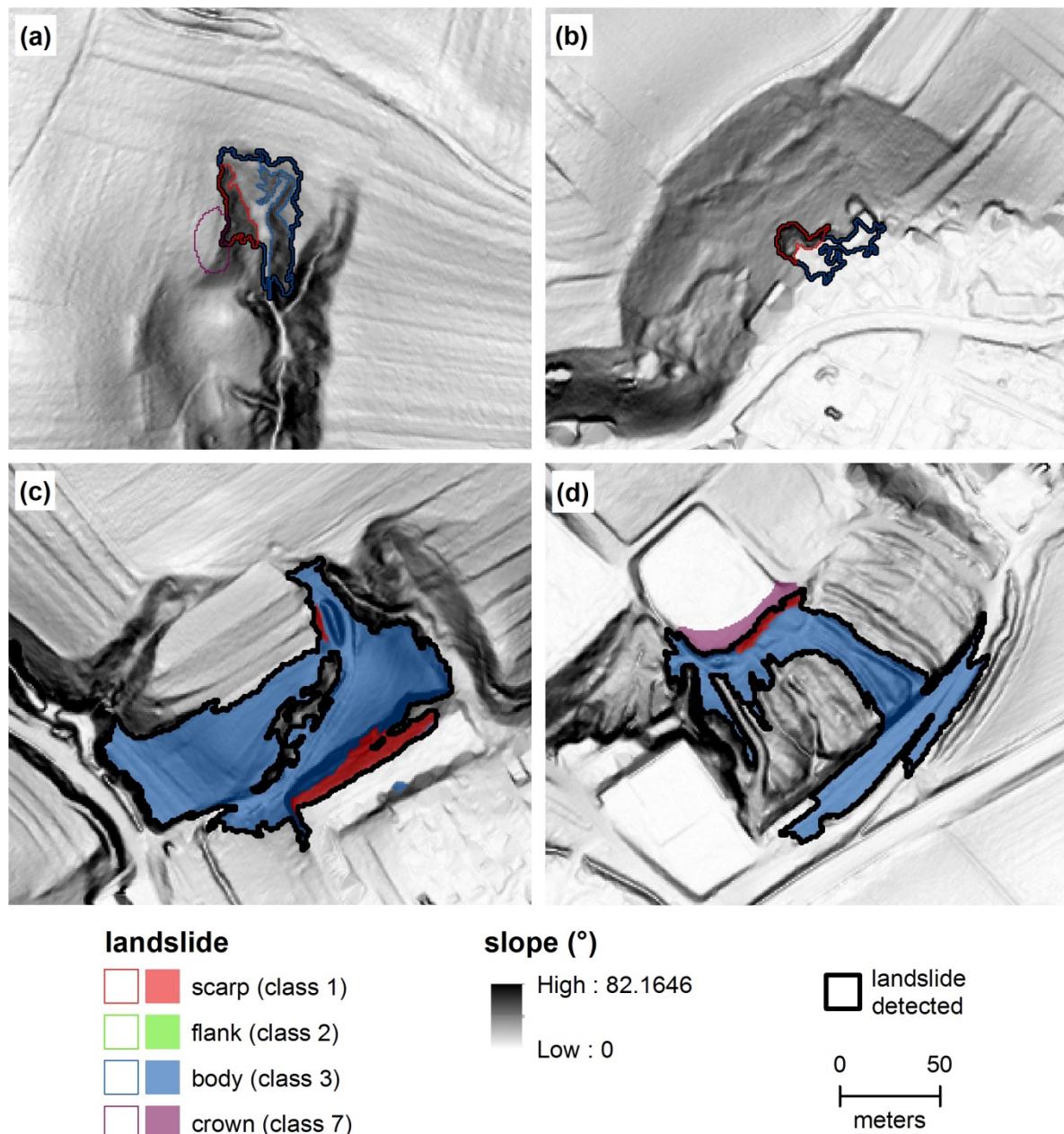
### 6.1 Landslide Detection

The object-oriented and open-source approach using only LiDAR derivatives allowed the detection of approximately 68.89% of landslide area (Table 5-8), and 65.54% of the total number of landslides with respect to an expert-based inventory (Table 5-9). However, compared to the accuracies achieved by MARTHA et al. (2011) and VAN DEN EECKHAUT et al. (2012) (s. 2.2), the results in this study are less accurate: the difference to the highest accuracy for the correctly detected total number of landslides is 7.76%-points related to the study of MARTHA et al. (2011:4938), and 35.86%-points considering scarp and body with a 50% threshold related to the study of VAN DEN EECKHAUT et al. (2012:39). The accuracy for the correctly identified area differs by 8.81%-points with respect to MARTHA et al. (2011:4938). The positional mismatch of 74.29% is 4.29%-points more than the range resulted in the study of VAN DEN EECKHAUT et al. (2012:38) for different expert-based inventories (50-70%). Moreover, the true-positive rates of 43.36% for correctly identified landslides, of 50.99% for all scarps, and of 61.47% for scarps in the inventory are quite low (i.e. 28.53%-points less for scarps in the inventory than VAN DEN EECKHAUT et al. (2012:38)).

The low true-positive rates might be ascribed to the high number of forms with similar characteristics to a scarp: For example a slope next to a natural eroded area by a river (Figure 6-1 a), a steep slope segment next to rough artificial surfaces (Figure 6-1 b), or a road ditch beside an area characterized by agricultural roads, drills or ditches (Figure 6-1 c-d). Even if those phenomena were mostly small-scaled, it was not possible to filter them by the algorithm, since the areas of the correctly identified landslides were also small with respect to the inventory (s. Figure 5-7). The high amount of small correctly classified landslides resulted from a combination of potentially too restrict queries and the *NeighborGrowing* algorithm. The growth of landslides beginning from the scarp stopped when no other landslide parts in the neighborhood were available. For example, if a filtered street crossed a landslide area, it was possible that the landslide areas became disconnected. Thus, one single inventory landslide could be encompassed by multiple classified landslides (Figure 6-2 d).

VAN DEN EECKHAUT et al. (2012) as well as MARTHA et al. (2011) differentiated in their accuracy assessment between shallow and deep-seated landslides. The discrepancy between the accuracies, especially in the study of VAN DEN EECKHAUT et al. (2012) was very high: 71% accuracy for deep seated landslides (scarp and body, total number: 38) and only 18.2% accuracy for shallow landslides (scarp and body, total number: 11) (VAN DEN EECKHAUT et al. 2012:38). Therefore, VAN DEN EECKHAUT et al. (2012) concluded that their approach, which has been adapted in this thesis, was suitable for the classification of landslides having a shear plane deeper than 3 m (VAN DEN EECKHAUT et al. 2012:38). Unfortunately, the invento-

ry used in this study had no information on the differentiation between shallow and deep-seated landslides that might partly explain the low detection rate of this study.



a: misclassification in natural eroded area. b: misclassification in artificial surfaces. c: misclassification due to agricultural roads. d: misclassification due to drills and ditches.

Figure 6-1: False-positive landslides (OWN REPRESENTATION).

Besides, by examining the first segmentation and the resulting scarps in *L1\_final* in relation to the inventory landslides, already 25 of the 148 landslides (16.89%) had no clear scarp segment, and additionally 22 landslides (14.86%) had a clear scarp segment but were not recognized as such. That means, already after the first segmentation level 47 landslides were missed (31.76%). The high number of missing landslides through missing scarps can be attributed to three main reasons: 1. the segmentation was not applicable or was too

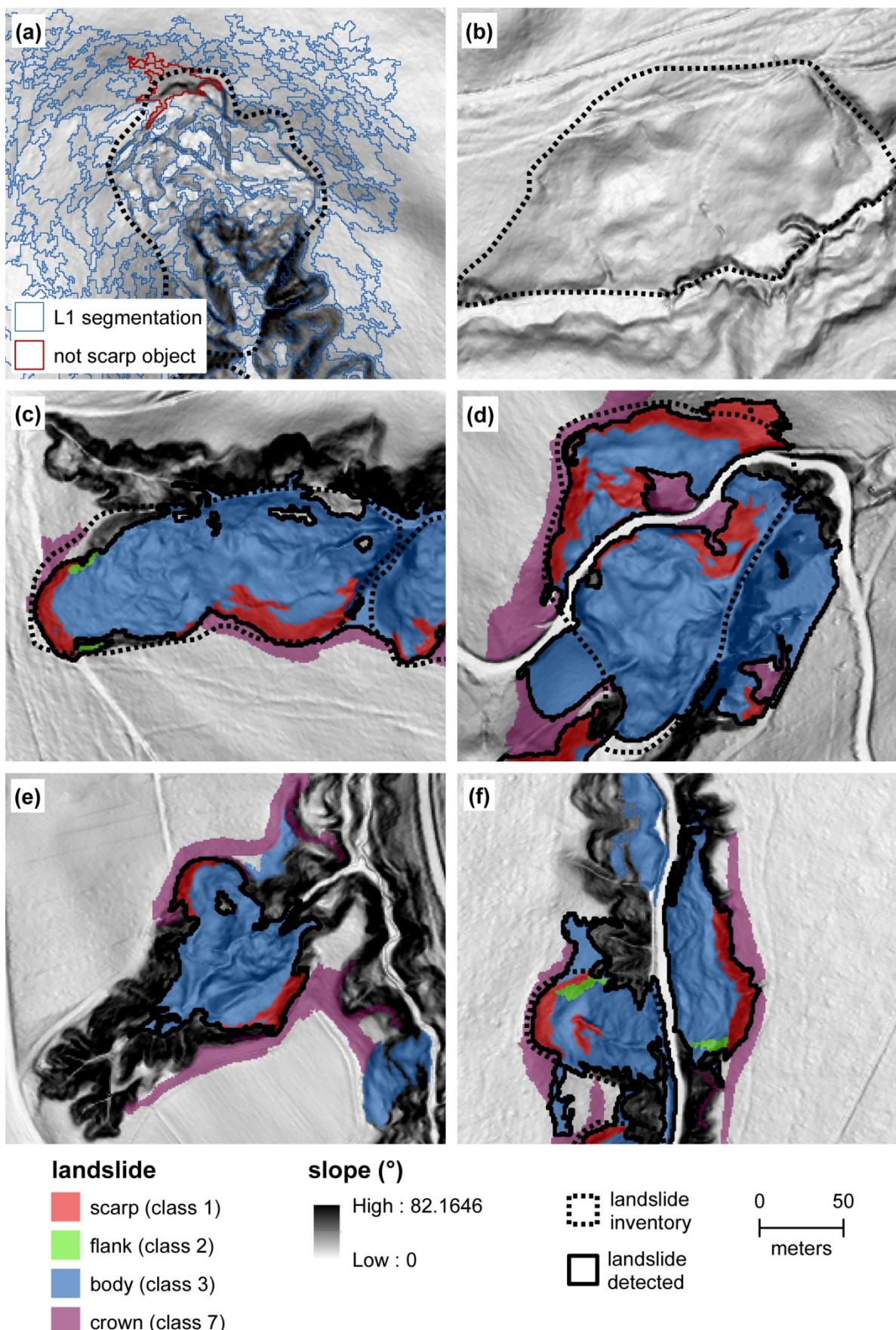
coarse to get a suitable scarp object (Figure 6-2 a), 2. the inventory landslide does not have a clear visible scarp (Figure 6-2 b), or 3. the query for the selection was too restrictive.

Due to the detection algorithm, missing the scarp means directly missing the complete landslide. Furthermore, possible landslide bodies resulted from the intersection between a segmentation and a bounding box calculated in flow direction of a scarp. The settings have a significant impact on the size of the bounding box, and thus also on the selection of bodies. To overcome these limitations, VAN DEN EECKHAUT et al. (2012) therefore proposed a probabilistic approach in which the different landslide parts are attributed with probabilities. Based on that additional information an evaluation can be performed, for example to find a missed scarp for an object with a high probability of being a landslide body or vice versa (VAN DEN EECKHAUT et al. 2012:40).

Moreover, the classification of landslides using data-driven threshold based on the k-means thresholding method was really challenging: On the one hand, the enormous number of parameters which had to be considered (textures, shape metrics, neighborhoods, land surface parameters) made the query, or selection, respectively, very complex, and thus reduce the possibility to easily transfer this approach to other regions. And on the other hand, for highly skewed data such as variance<sub>Flow</sub> or length-width-ratio, the resulting thresholds were impractical due to large gaps between cluster centers. In such cases, additional thresholds were computed limiting the input data to the first or third quartile value as minimum or maximum value, respectively. Additionally, it was a time-consuming process to find “good” thresholds for the separability of possible false-positive candidates.

Therefore, the author recommends using (un)supervised classifiers such as random forest, or supervised vector machines for the classification instead, even if this requires interaction with the created objects through the selection of training data.

The results of the objective function were a good support for finding optimal segmentation scale parameters. However, in the case of multiple peaks, the selection remains still subjective. To give a desirable fine, medium, or coarse-scale segmentation more weight in the objective function, JOHNSON et al. (2015) proposed to combine the OF with the F-measure (JOHNSON et al. 2015:2295). This suggestion could not be implemented, yet. Besides, with respect to the 25 missed inventory landslides due to no clear scarp segments in L1 (see above), it is questionable if the (seeded) region growing segmentation is a suitable application for the detection of scarps. VAN DEN EECKHAUT et al. (2012) used a contrast split segmentation to detect possible scarps, but this algorithm was not found in the used open-source products (VAN DEN EECKHAUT et al. 2012:33). Here, further investigation can be made on the adaptive thresholding function *thresh()* in the *EBImage* package to fulfill this task.



a: missed landslide due to coarse segmentation. b: missed landslide due to missing clear scarp characteristic. c-d: detected inventory landslide. e-f: detected landslide outside of inventory.

Figure 6-2: Examples of (not) detected landslides (OWN REPRESENTATION).

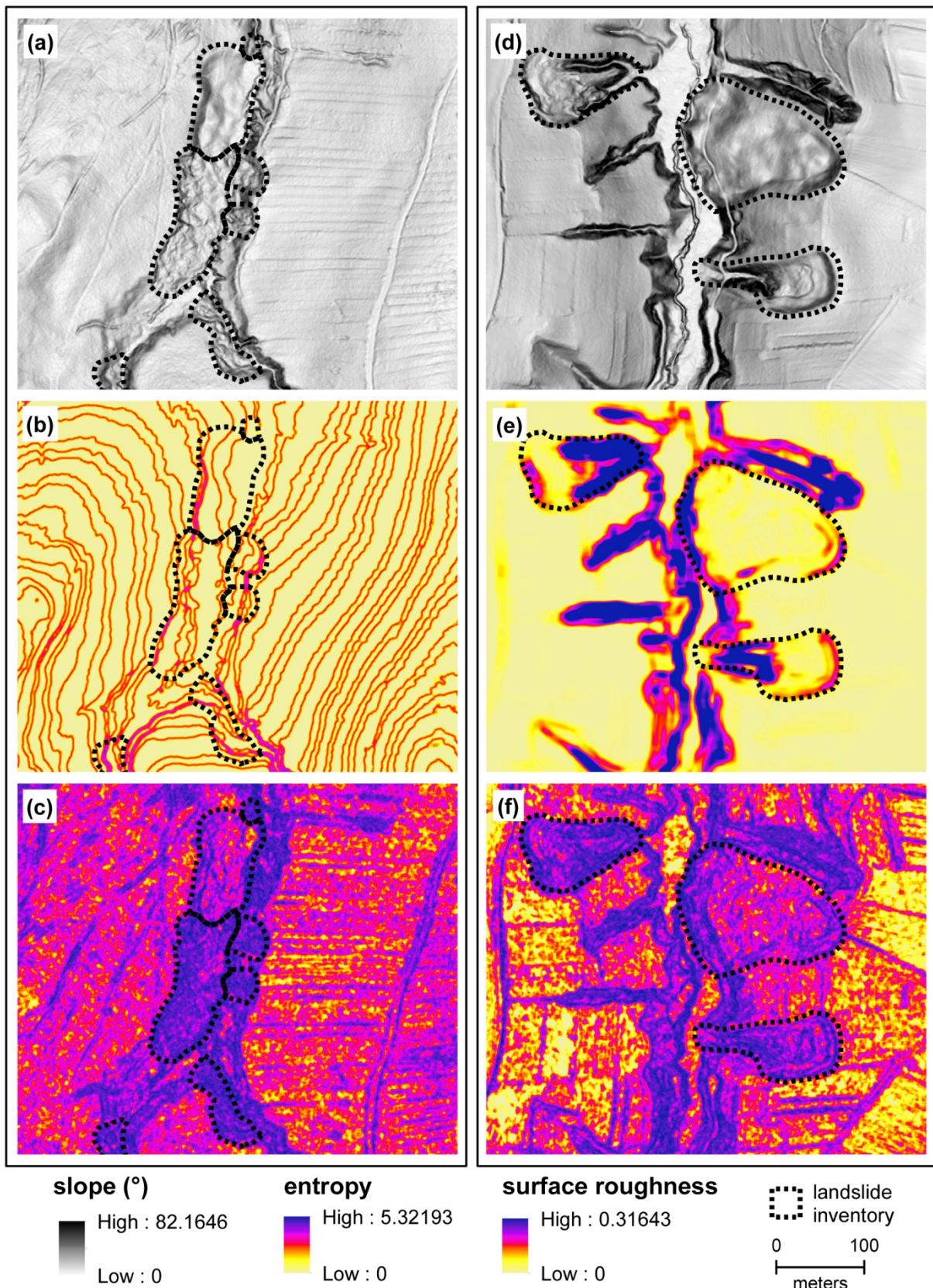
Nevertheless, the algorithm allowed also the detection of 22.31 ha of additional landslide area (total number: 62) outside of the inventory (s. Table 5-8), which shows the great potential of the algorithm to support and enhance the manual mapping of landslides. In addition, 95.71% of the correctly classified landslides were located under forest and semi natural areas where images from passive optical sensors are unusable. Furthermore, the classification results support the statement that geomorphological or historical inventories are never complete (GUZZETTI 2004:2), and are based on the subjective valuation of the experts. Of course, this also applies to the accuracy assessment of this thesis, which was done only by the author and based on the DTM derivatives.

In the study of PETSCHKO et al. (2016) the landslide mapping rate was approximately 51,800 ha mapped per person month (working 40 h/week) for the manual mapping of landslides in form of polygons (PETSCHKO et al. 2016:866). It is assumed that by the support of an automatic detection, a faster landslide mapping rate could be achieved (MONDINI et al. 2011:1755).

## 6.2 Input Parameters

The performance of different land surface parameters, shape indices and textural features were investigated for the landslide processing.

The land surface parameters as well as the textural features were primarily based on the DTM extracted from LiDAR data. However, according to LIU (2008) the processing of the LiDAR point cloud and the terrain modelling for large forested areas are challenging tasks, and artefacts, especially in complex and highly variable morphology, are possible (LEE & YOUNAN 2003, LIU 2008:35). However, in the frame of this thesis no investigations could be made on finding or correcting irregularities in the DTM. Therefore, it was possible that errors could also propagate in other DTM derivatives and, thus lead to misclassifications. Besides, according to HENGL (2006) and following the Nyquist frequency concept (SHANNON 1949) the grid resolution of the DTM should not exceed half the average shortest distance of point pair samples from that a DTM is interpolated (HENGL 2006:1289). Based on that concept VAN DEN EECKHAUT et al. (2012) decided to work with 2 m resolution for example (VAN DEN EECKHAUT et al. 2012:33). However, there was no information on how many samples were taken, or what the average shortest distance was, for the TerraModeler (s. 3.2). Therefore, the analysis was performed with 1 m resolution.



a, d: slope. b: entropy on DTM. c, f: entropy on slope. e: surface roughness. a-c: comparison of entropy results depending on texture input. d-f: comparison of surface roughness and entropy.

Figure 6-3: Difficulties with input parameters (OWN REPRESENTATION).

Related to the land surface parameters the slope was the key feature in all classification steps. Slope or slope edges were substantial parts in all segmentations and nearly all que-

ries. Besides, openness, sky-view factor and normalized height served well for the segregation of false-positive scarp and body candidates, such as road ditches or river banks in downslope area, or agricultural fields, respectively. The application of the maximum curvature according to the study of TAROLLI et al. (2012) contributed greatly to the restriction of the potential scarps in L1, and the detection of crowns in L2. However, in contrast to the study of VAN DEN EECKHAUT et al. (2012) the surface roughness in form of the vector ruggedness measure showed less potential to successfully separate landslide bodies from non-landslide bodies in L3 (Figure 6-3 d-f, VAN DEN EECKHAUT et al. (2012) used the topographic ruggedness index by RILEY et al. (1999) for this purpose). Different surface roughness algorithms calculated with different window-sizes using high resolution data were already investigated as measurement for the identification of active landslides, or as proxy for landslide age by BERTI et al. (2013), and GOETZ et al. (2014), respectively. GOETZ et al. (2014) found only weak correlations between surface roughness and landslide age (best Spearman's rho 0.58) (GOETZ et al. 2014:1701ff.). Moreover, BERTI et al. (2013) discovered a drop of 15% in accuracy (from 90% to 75%) by considering landslides under forest area in their prediction of active slopes (BERTI et al. 2013:17). They explained the decrease by the inherent roughness of forest areas, and thus the loose of discriminatory capability of the surface roughness algorithms in their model (BERTI et al. 2013:14). Considering that, around 90% of the landslides in the inventory were mapped under forest area (s. Table 3-2) and circa 68% of the area consists of forest area (s. Table 3-1), this could be an explanation for the low capability of separability by the surface roughness.

Fortunately, the textural features – especially the entropy, calculated in flow direction, according to the approach of STUMPF & KERLE (2011), provided good values for characterizing the landslide body parts (Figure 6-3 c, f). Commonly, the texture is computed on a DTM input. However, textures based on the DTM did not show up in wide parts of the image. Instead, using the slope as input, clear textures were visible (Figure 6-3 a-c). The author assumes that this phenomenon is due to the rescaling of the input to gray level (0 to 255), and the resulting loss of information during that process.

Related to the shape metrics, especially the shape index and the length-width-ratio proved to be very effective for unselecting long, narrow, and straight objects often representing false-positive candidates of scarps. In addition, the function *MainDirection* was an extremely helpful and performant tool to determine the orientation of an object in space.

Apart from the shape metrics, the computation of nearly all parameters is primarily based on a moving-window approach. The question of the suitable setting, is therefore always a question of scale and size of the investigated objects (DRĀGUĀ & EISANK 2011:183f.). It is a desirable, but also challenging task, to create a theoretical model to determine optimal window

sizes in dependence of the application (ZHOU et al. 2008:78). It is even more challenging, when considering that objects are derived based on these optimal settings. In the frame of the thesis, no optimal settings for all input parameters used could be found in literature, or even investigated in this study. Therefore, the selection of suitable window sizes was based on the subjective assessment of the author (Table 4-2). Moreover, BLASCHKE et al. (2004) stated that objects will never be optimal or meaningful at all scales, or for any phenomena. This is due to the fact that in nature hard boundaries are not existing, but rather clear or soft transitions in land cover (BLASCHKE et al. 2004:220).

### 6.3 Usability of Open Source Products and *Lslide* Package

The integration of SAGA GIS, GRASS GIS, TauDEM, and Relief Visualization Toolbox in the free development environment R could be easily fulfilled by `system()`-commands, or by the corresponding packages (*RSAGA*, *rgrass7*), respectively. Moreover, R comprises many spatial packages that contribute to the development of the detection algorithm and the functions in the *Lslide* package. SAGA and GRASS GIS provide basic terrain analysis modules, and in combination, they build a good fundament for this thesis. Generally, these modules can be easily accessed, extended or combined in R by corresponding functions, which make them a powerful tool. The documentary of the modules in SAGA GIS is quite sparsely in comparison to GRASS GIS. However, both have a large user community, mailing lists, and forums, so in the case of a question one can quickly get help.

However, some characteristics of the open source products are still challenging and unhandy. Before using the functions in the created *Lslide* package, all software must be installed individually by the user. Currently, *RSAGA* (version 0.94-5) only supports SAGA GIS versions from 2.0.4 to 2.2.3. However, since 20-03-2017 the SAGA GIS version 4.0.1 has already been released, which means that for this application an older version must be installed that does not have bug fixes of the following versions. Besides, GRASS GIS has its own logical database structure. Before accessing any GRASS GIS module, therefore at the beginning of every R session a location and mapset must be initialized in that imported or processed files are stored. Hence, to access files in GRASS GIS, they must be imported first, and to use processed files from GRASS GIS in R or SAGA GIS, they must be exported first. In addition, raster calculations in SAGA GIS are saved in the typical SAGA format (`.sdat` or `.sgrd`) which can only be read into R by the *rgdal* package (BIVAND et al. 2016). Furthermore, the extent of results of some algorithms that are based on a moving-window approach, are smaller than the input. That leads to undesirable loss of information at the border and to strange statistical values for border objects using zonal statistics. Therefore, it is suggested to calculate all parameters first and then limit the study area to the smallest extent result. Moreover, the run-time of algorithms with similar objective such as the segmentation (s.

4.3.1), can strongly differ from one software to another, and often to the disadvantage of GRASS GIS. Therefore, an investigation of similar modules and their performance is recommended.

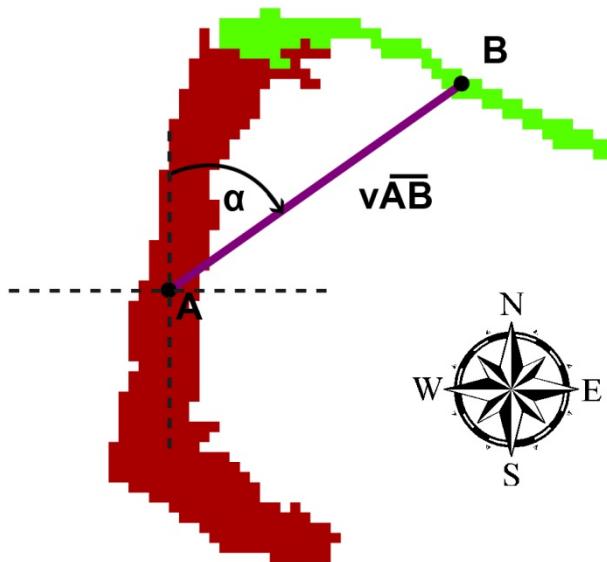
The possibilities of working with spatial data in R are manifold, and mostly well documented by vignettes. However, to load a shapefile with more than 1,000,000 objects (i. e. segmentation in L1) into R the corresponding functions in the *sp*, *maptools*, *shapefile*, or *rgdal* packages took nearly 24 hours, or even an error occurred due to memory limits of the author's computer. Fortunately, the enormous size of data could be handled within several minutes with functions of the new *sf* package (here version 0.3-4, from 06-02-2017, first release: 26-10-2016).

The implemented functions in *Lslide* showed reasonable performance in relation to the size of data that must be executed (s. 4.3). However, there is still potential to speed up the processes: In most functions, a simple for-loop was implemented for iterating the data. Instead, functions of the apply-family or parallel processing for example by the *foreach* package (REVOLUTION ANALYTICS & WESTON 2015) could be used.

Nonetheless, especially, the functions for neighbor operations (s. 4.3.5) could greatly contribute to the detection of landslides. The statistic parameters of "weighted mean", "mean difference to", and "relative border to" were very effective for the separability of objects. However, in the frame of the thesis the function results in *Lslide* were mostly validated by visual examination, or example data sets. Thus, there is no guarantee of complete correctness of the algorithms.

The function *getBoundingBox* was used as a tool to attribute objects lying in a direction of other specific objects, such as landslide bodies lying in flow direction of the scarps. By the intersection of the created rectangles with their environment, there is always the possibility to select objects lying in inverse direction. This is because, the bounding box is initialized from the object's centroid to that it belongs (s. 4.3.3). Even if there are function parameters controlling the position of the initial point of the bounding box, due to the shape irregularity of objects a creation of a bounding box without inverse direction intersections is very challenging. To overcome this limitation and to filter inverse direction intersections, the author suggests the following approach which is not implemented in the *Lslide* package, yet (Figure 6-4): (1) For every object a point *P* on its surface is created (i.e. point *A* and *B*). Here, the usage of the centroid can lead to erroneous values for objects with a low convexity value. (2) For every object point *P* a line vector is created to its neighbor object points *Pneighbors* (i.e. vector  $\nu\overrightarrow{AB}$ ). (3) The direction  $\alpha$  is then calculated by trigonometrical functions such as *atan2* in R. (4) Comparison of the initial object direction (i.e. flow direction of a specific object) to

the directions of the object neighbors to this object (i.e.  $\alpha$ ), and unselecting of wrongly attributed objects if necessary.



A, B: point on the object's surface.  $v\bar{AB}$ : line connecting point A with B.  $\alpha$ : direction of line related to cardinal directions. N: 0|360°, E: 90°, S: 180°, W: 270°.

**Figure 6-4: Creation of directional neighborhood (own REPRESENTATION).**

Besides, the execution of the objective function was the most computationally time-consuming process. Fortunately, it was possible to successfully outsource this process on an external server.

## 7 Conclusion and Perspective

In this thesis an object-oriented approach was developed to automatically detect landslides based on only LiDAR DTM derivatives and open-source software. For the detection multiple queries, based on thresholds derived by a k-means clustering method, were applied on the data using a three level step approach. The algorithm could successfully be implemented in the free development environment R and encompasses mainly functions, or modules of SAGA GIS, GRASS GIS, and R packages. In addition, different parameters such as land surface parameters, shape metrics, and textural features were investigated on their capability to effectively characterize the morphometric properties of the different landslide parts.

The results led to landslide detection accuracies of approximately 68.89% in correctly identified landslide areas, and 65.54% in the correctly recognized total number of inventory landslides by considering scarp, flank and landslide bodies. Around 95.71% of the correctly detected landslide area were located on forest and semi natural areas, where the detection based on images from passive optimal remote sensing are limited. The true-positive rate of only 43.36% correctly identified landslides was quite low due to many false-detected small-scale landslides. However, the additional detection of 22.31 ha of landslide area outside the inventory – an additional of 12.47% with respect to the total area of the inventory, shows the great potential of the algorithm to support the manual mapping method.

Due to the limited time of this master thesis, there is still a lot of research to be done:

- Testing the performance of the presented algorithm on a different study area with different geological settings.
- Creation of a function in *Lslide* that automatically detects landslide parts in given landslide polygons of an inventory.
- Investigation of the impact of different grid resolutions (2 x 2 m, 3 x 3 m, ...) on the resulted accuracy to find an “optimal” scale.
- Comparison of the existing object-oriented and pixel-based algorithms using the same study area.
- Testing and finding descriptive and robust statistic parameters for the different landslide parts (scarp, body, flank) using textural features and land surface parameters calculated with different window sizes (referring to the study of TAROLLI et al. (2012)).

The automatized detection algorithms of landslides can greatly fasten up and objectively support the creation of landslide inventories. This way they can contribute to a better performance in susceptibility mapping, and thus to a better understanding of the linkage between landslides and climate change. However, manual verifications of the results or field works are still necessary.

## 8 References

- ACHANTA, R., SHAJI, A., SMITH, K. , LUCCHI, A., FUA, P. & S. SUSSTRUNK (2012): SLIC superpixels compared to state-of-the-art superpixel methods. – IEEE transactions on pattern analysis and machine intelligence 34, 11, 2274–2282.
- ADAMS, R. & L. BISCHOF (1994): Seeded region growing. – IEEE transactions on pattern analysis and machine intelligence 16, 6, 641–647.
- BAATZ, M., HOFFMANN, C & G. WILLHAUCK (2008): Progressing from object-based to object-oriented image analysis. In: BLASCHKE, T., LANG, S. & G. HAY (eds.): Object-based image analysis: Spatial concepts for knowledge-driven remote sensing applications. Lecture notes in geoinformation and cartography. Berlin: Springer, 29–42.
- BECHTEL, B., RINGELER, A. & J. BOEHNER (2008): Segmentation for Object Extraction of Trees using MATLAB and SAGA. – SAGA–Seconds Out, Hamburger Beiträge zur Physischen Geographie und Landschaftsökologie, 19, 59–70.  
[https://www.geo.uni-hamburg.de/geographie/dokumente/personen/publikationen/bechtel/bechtel\\_et\\_al\\_2008.pdf](https://www.geo.uni-hamburg.de/geographie/dokumente/personen/publikationen/bechtel/bechtel_et_al_2008.pdf) (last access: 17-04-2017).
- BELL, R., PETSCHEKO, H., RÖHRS, M. & A. DIX (2012): Assessment of landslide age, landslide persistence and human impact using airborne laser scanning digital terrain models. – Geografiska Annaler: Series A, Physical Geography 94, 1, 135–156.
- BERTI, M., CORSINI, A. & A. DAEHNE (2013): Comparative analysis of surface roughness algorithms for the identification of active landslides. – Geomorphology 182, 1–18.
- BINS, L.S., FONSECA, L.M.G., ERTHAL, G.J. & F.M. II (1996): Satellite imagery segmentation: a region growing approach. – Simpósio Brasileiro de Sensoriamento Remoto 8, 677–680.
- BIVAND, R.S. (2011): Geocomputation and Open Source Software: Components and Software Stacks. – SSRN Electronic Journal, 1–37.
- BIVAND, R.S., KEITT, T. & B. ROWLINGSON (2016): rgdal: Bindings for the Geospatial Data Abstraction Library. R package version 1.2-4.  
<https://CRAN.R-project.org/package=rgdal> (last access: 04-09-2016).
- BIVAND, R.S. & N. LEWIN-KOH (2016): maptools: Tools for Reading and Handling Spatial Objects: R package version 0.8-40.  
<https://CRAN.R-project.org/package=maptools> (last access: 29-03-2017).
- BIVAND, R.S. & G. PIRAS (2015): Comparing Implementations of Estimation Methods for Spatial Econometrics. – Journal of Statistical Software 63, 18, 1–36.

- BIVAND, R.S. & C. RUNDEL (n.y.): rgeos: Interface to Geometry Engine - Open Source (GEOS). R package version 0.3-21.  
<https://CRAN.R-project.org/package=rgeos> (last access: 29-03-2017).
- BIVAND, R.S. (2016): rgrass7: Interface Between GRASS 7 Geographical Information System and R.  
<https://CRAN.R-project.org/package=rgrass7> (last access: 02-01-2017).
- BIVAND, R.S., PEBESMA, E. & V. GÓMEZ-RUBIO (2013<sup>2</sup>): Applied Spatial Data Analysis with R. Use R! New York: Springer.
- BLASCHKE, T. (2010): Object based image analysis for remote sensing. – ISPRS Journal of Photogrammetry and Remote Sensing 65, 1, 2–16.
- BLASCHKE, T., BURNETT, C. & A. PEKKARINEN (2004): Image Segmentation Methods for Object-based Analysis and Classification. In: VAN DER MEER, F.D., ABRAMS, M., CURRAN, P., DEKKER, A., JONG, S.M.D. & M. SCHAEPMAN (eds.): Remote Sensing Image Analysis: Including The Spatial Domain. Remote Sensing and Digital Image Processing. Dordrecht: Springer, 211–236.
- BLASCHKE, T., FEIZIZADEH, B. & D. HOLBLING (2014a): Object-Based Image Analysis and Digital Terrain Analysis for Locating Landslides in the Urmia Lake Basin, Iran. – IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 7, 12, 4806–4817.
- BLASCHKE, T., HAY, G.J., KELLY, M., LANG, S., HOFMANN, P., ADDINK, E., QUEIROZ FEITOSA, R., VAN DER MEER, F., VAN DER WERFF, H., VAN COILLIE, F. & D. TIEDE (2014b): Geographic Object-Based Image Analysis – Towards a new paradigm. – ISPRS Journal of Photogrammetry and Remote Sensing 87, 180–191.
- BLASCHKE, T., LANG, S. & G. HAY (2008): Object-based image analysis: Spatial concepts for knowledge-driven remote sensing applications. Lecture notes in geoinformation and cartography. Berlin: Springer.
- BLONDEL, E. (2016): cleangeo: Cleaning Geometries from Spatial Objects. R package version 0.2-1.  
<https://CRAN.R-project.org/package=cleangeo> (last access: 29-03-2017).
- BÖHNER, J., SELIGE, T. & A. RINGELER (2006): Image segmentation using representativeness analysis and region growing. – Göttinger Geographische Abhandlungen, 115, 29–38.
- BOOTH, A.M., ROERING, J.J. & J.T. PERRON (2009): Automated landslide mapping using spectral analysis and high-resolution topographic data: Puget Sound lowlands, Washington, and Portland Hills, Oregon. – Geomorphology 109, 3-4, 132–147.

- BOYCE, R.R. & W.A.V. CLARK (1964): The Concept of Shape in Geography. – Geographical Review 54, 4, 561–572.
- BRENNING, A. (2008): Statistical Geocomputing combining R and SAGA: The example of landslide susceptibility analysis with generalized additive Models. [<http://mirror.ufs.ac.za/sagagis/SAGA%20-%20Documentation/HBPL19/hbpl19\\_03.pdf>](http://mirror.ufs.ac.za/sagagis/SAGA%20-%20Documentation/HBPL19/hbpl19_03.pdf) (last access: 14-01-2017).
- BRITISH GEOLOGICAL SURVEY (n.y.): The classification of landslides by the BGS: Based on Varnes 1978 and adapted by Cruden and Varnes 1996, redrawn by BGS/NERC. [<http://www.bgs.ac.uk/research/engineeringGeology/images/landslide\\_types\\_large.jpg>](http://www.bgs.ac.uk/research/engineeringGeology/images/landslide_types_large.jpg) (last access: 16-04-2017).
- CAMPBELL, J.B. & R.H. WYNNE (2011<sup>5</sup>): Introduction to remote sensing. New York: Guilford Press.
- CONRAD, O., BECHTEL, B. , BOCK, M., DIETRICH, H., FISCHER, E., GERLITZ, L., WEHBERG, J. , WICHMANN V. & J. BÖHNER (2015): System for Automated Geoscientific Analyses (SAGA) v. 2.1.4. – Geoscientific Model Development 8, 7, 1991–2007.
- CRUDEN, D.M. & D.J. VARNES (1996): Landslide Types and Processes. – Special Report Transportation Research Board, National Academy of Sciences 247, 36–75.
- DIETRICH, H. & J. BÖHNER (2008): Cold Air Production and Flow in a Low Mountain Range Landscape in Hessia. – SAGA–Seconds Out, Hamburger Beiträge zur Physischen Geographie und Landschaftsökologie 19, 37–48.
- DIKAU, R. (1989): The application of a digital relief model to landform analysis in geomorphology. In: RAPER, J. (ed.): Three dimensional applications in geographical information systems. London: Taylor & Francis, 51–77.
- DIKAU, R. (1990): Geomorphic landform modelling based on Hierarchy Theory. In: BRASSEL K. (ed.). Proceedings 4<sup>th</sup> International Symposium on Spatial Data Handling. Zürich, 230–239.
- DOGLIONI, A. & V. SIMEONE (2014): Geomorphometric analysis based on discrete wavelet transform. – Environmental Earth Sciences 71, 7, 3095–3108.
- DRĂGUȚ, L., TIEDE, D. & S.R. LEVICK (2010): ESP: A tool to estimate scale parameter for multi-tiresolution image segmentation of remotely sensed data. – International Journal of Geographical Information Science 24, 6, 859–871.
- DRĂGUȚ, L., CSILLIK, O., EISANK, C. & D. TIEDE (2014): Automated parameterisation for multi-scale image segmentation on multiple layers. – ISPRS Journal of Photogrammetry and Remote Sensing 88, 119–127.

- DRĂGUȚ, L. & C. EISANK (2011): Object representations at multiple scales from digital elevation models. – *Geomorphology* 129, 3-4, 183–189.
- ESPINDOLA, G.M., CAMARA, G., REIS, I.A., BINS, L.S. & A.M. MONTEIRO (2006): Parameter selection for region-growing image segmentation algorithms using spatial autocorrelation. – *International Journal of Remote Sensing* 27, 14, 3035–3040.
- FRALEY, C. & A.E. RAFTERY (2002): Model-based Clustering, Discriminant Analysis, and Density Estimation. – *Journal of the American statistical Association* 97, 158, 611–631.
- GALLANT, J.C. & J.P. WILSON (2000): Primary Topographic Attributes. In: WILSON, J.P. & J.C. GALLANT (eds.): *Terrain analysis: Principles and applications*. New York: Wiley, 51–86.
- GLENN, N.F., STREUTKER, D.R., CHADWICK, D.J., THACKRAY, G.D. & S.J. DORSCH (2006): Analysis of LiDAR-derived topographic information for characterizing and differentiating landslide morphology and activity. – *Geomorphology* 73, 1-2, 131–148.
- GOETZ, J.N., BELL, R. & A. BRENNING (2014): Could surface roughness be a poor proxy for landslide age? Results from the Swabian Alb, Germany. – *Earth Surface Processes and Landforms* 39, 12, 1697–1704.
- GRUBER, S. & S. PECKHAM (2009): Land-Surface Parameters and Objects in Hydrology. In: HENGL, T. & H.I. REUTER (eds.): *Geomorphometry: Concepts, software, applications* 33. *Developments in Soil Science* 33. Amsterdam: Elsevier, 171–194.
- GUZZETTI, F. (2004, October, 7-8): Landslide Mapping, Hazard Assessment and Risk Evaluation, Limits and Potential: Proceeding of International Symposium on Landslide and Debris Flow Hazard Assessment.
- GUZZETTI, F., MONDINI, A.C., CARDINALI, M., FIORUCCI, F., SANTANGELO, M. & K.-T. CHANG (2012): Landslide inventory maps: New tools for an old problem. – *Earth-Science Reviews* 112, 1-2, 42–66.
- HARALICK, R.M., SHANMUGAM, K. & I. DINSTEIN (1973): Textural Features for Image Classification. – *IEEE Transactions on Systems, Man, and Cybernetics SMC-3*, 6, 610–621.
- HAY, G.J. & G. CASTILLA (2008): Geographic Object-Based Image Analysis (GEOBIA): A new name for a new discipline. In: BLASCHKE, T., LANG, S. & G. HAY (eds.): *Object-based image analysis: Spatial concepts for knowledge-driven remote sensing applications. Lecture notes in geoinformation and cartography*. Berlin: Springer, 75–89.
- HENGL, T. (2006): Finding the right pixel size. – *Computers & Geosciences* 32, 9, 1283–1298.

- HIGHLAND, L. & P.T. BOBROWSKY (2008): The landslide handbook: A guide to understanding landslides. U.S. Geological Survey.
- HIJMANS, R.J. (2016): raster: Geographic Data Analysis and Modeling. R package version 2.5-8.  
[<https://CRAN.R-project.org/package=raster>](https://CRAN.R-project.org/package=raster) (last access: 29-03-2017).
- HÖFLE, B. & M. RUTZINGER (2011): Topographic airborne LiDAR in geomorphology: A technological perspective. – *Zeitschrift für Geomorphologie* 55, 2, 1–29.
- HUNGR, O., LEROUEIL, S. & L. PICARELLI (2014): The Varnes classification of landslide types, an update. – *Landslides* 11, 2, 167–194.
- INTERNATIONAL FEDERATION OF RED CROSS AND RED CRESCENT SOCIETIES (2015): World disasters report 2015: Focus on local actors, the key to humanitarian effectiveness. Geneva, Switzerland.
- JABOYEDOFF, M., OPPIKOFER, T., ABELLÁN, A., DERRON, M.-H., LOYE, A. , METZGER, R. & A. PEDRAZZINI (2012): Use of LIDAR in landslide investigations: A review. – *Natural Hazards* 61, 1, 5–28.
- JAKOB, M. & S. LAMBERT (2009): Climate change effects on landslides along the southwest coast of British Columbia. – *Geomorphology* 107, 3-4, 275–284.
- JOHNSON, B., BRAGAIS, M., ENDO, I., MAGCALE-MACANDOG, D. & P. MACANDOG (2015): Image Segmentation Parameter Optimization Considering Within- and Between-Segment Heterogeneity at Multiple Scale Levels: Test Case for Mapping Residential Areas Using Landsat Imagery. – *ISPRS International Journal of Geo-Information* 4, 4, 2292–2305.
- KALBERMATTEN, M., VAN DE VILLE, D., TURBERG, P., TUIA, D. & S. JOOST (2012): Multiscale analysis of geomorphological and geological features in high resolution digital elevation models using the wavelet transform. – *Geomorphology* 138, 1, 352–363.
- KAMP, N.: Using High-Resolution Airborne LiDAR-Data for Landslide Mapping in the Eastern Alps. Thesis, University of Graz. Institute of Geography and Regional Science. Graz, Austria.
- KUMAR, R.M. & K. SREEKUMAR (2014): A Survey on Image Feature Descriptors. – *International Journal of Computer Science and Information Technologies* 5, 6, 7668–7673
- LANG, S. (2008): Object-based image analysis for remote sensing applications: modeling reality – dealing with complexity. In: BLASCHKE, T., LANG, S. & G. HAY (eds.): Object-based image analysis: Spatial concepts for knowledge-driven remote sensing applications. Lecture notes in geoinformation and cartography. Berlin: Springer, 3–27.

- LANG, S. & T. BLASCHKE (2007): Landschaftsanalyse mit GIS. Stuttgart: Ulmer.
- LEE, H.S. & N.H. YOUNAN (2003): DTM extraction of Lidar returns via adaptive processing. – IEEE Transactions on Geoscience and Remote Sensing 41, 9, 2063–2069.
- LEITÄO, A.B., J. MILLER, J. AHERN & K. MCGARIGAL (2006): Measuring landscapes: A planner's handbook. Washington, DC: Island Press.
- LEOPOLD, P. (2012): Kriechbewegungen in den neogenen Beckensedimenten des Burgenlandes: Verbreitung, Vergleich natürlicher Einflussfaktoren, Bewegungsmechanismen und Auslöser. Dissertation, Universität Wien. Technische Geologie. Wien.
- LEOPOLD, P. (2016): Genauigkeitsschätzung des Inventars. e-mail correspondence with Philip Leopold, Scientist, AIT Austrian Institute of Technology GmbH (30-11-2016).
- LEOPOLD, P., DRAGANITS, E., HEISS, G. & E. KOVACS (2013): A Geotechnical Explanation for the Transition from Creep to Slides in the Alpine Foreland. In: MARGOTTINI, C., CANUTI, P. & K. SASSA (eds.): Landslide Science and Practice 4: Global Environmental Change. Dordrecht: Springer.
- LEOPOLD, P., OBLIN, A. & G. HEISS (2015): Forschungsprojekt Massenbewegungen im Bezirk Oberpullendorf: Neubearbeitung der Gefahren-Hinweiskarte sowie Empfehlungen zur Flächenwidmung. Wien.
- LIN, C.-W., TSENG, C.-M., TSENG, Y.-H., FEI, L.-Y., HSIEH, Y.-C. & P. TAROLLI (2013): Recognition of large scale deep-seated landslides in forest areas of Taiwan using high resolution topography. – Journal of Asian Earth Sciences 62, 389–400.
- LIU, D. & J. Yu (2009): Otsu Method and K-means. – IEEE Computer Society, 344–349.
- LIU, X. (2008): Airborne LiDAR for DEM generation: Some critical issues. – Progress in Physical Geography 32, 1, 31–49.
- LU, P., STUMPF, A., KERLE, N. & N. CASAGLI (2011): Object-Oriented Change Detection for Landslide Rapid Mapping 8, 4, 701–705.
- MALAMUD, B.D., TURCOTTE, D.L., GUZZETTI, F. & P. REICHENBACH (2004): Landslide inventories and their statistical properties. – Earth Surface Processes and Landforms 29, 6, 687–711.
- MANETTI, L. & G. STEINMANN (2007): 3DeMoN ROBOVEC: integration of a new measuring instrument in anexisting generic remote monitoring platform. In: 7<sup>th</sup> international symposium on field measurements in geomechanics. Boston, 1–12.
- MANN, U., PRADHAN, B., PRECHTEL, N. & M.F. BUCHROITHNER (2012): An Automated Approach for Detection of Shallow Landslides from LiDAR Derived DEM Using Geomorpho-

- logical Indicators in a Tropical Forest. In: PRADHAN, B. & M. BUCHROITHNER (eds.): Terri-genous Mass Movements: Detection, Modelling, Early Warning and Mitigation Using Geoinformation Technology. Berlin: Springer, 1–22.
- MARTHA, T.R., KERLE, N., JETTEN, V., VAN WESTEN, C.J. & K.V. KUMAR (2010): Characterising spectral, spatial and morphometric properties of landslides for semi-automatic detection using object-oriented methods. – *Geomorphology* 116, 1-2, 24–36.
- MARTHA, T.R., KERLE,, N., VAN WESTEN, C.J., JETTEN, V. & K.V. KUMAR (2011): Segment Optimization and Data-Driven Thresholding for Knowledge-Based Landslide Detection by Object-Based Image Analysis. – *IEEE Transactions on Geoscience and Remote Sensing* 49, 12, 4928–4943.
- MCKEAN, J. & J. ROERING (2004): Objective landslide detection and surface morphology mapping using high-resolution airborne laser altimetry. – *Geomorphology* 57, 3–4, 331–351.
- MINER, A., FLENTJE, P., MAZENGARB, C. & D. WINDLE (2010): Landslide Recognition using LiDAR derived Digital Elevation Models-Lessons learnt from selected Australian examples. – Faculty of Engineering, 1–9.
- MONDINI, A.C., GUZZETTI, F., REICHENBACH, P., ROSSI, M., CARDINALI, M. & F. ARDIZZONE (2011): Semi-automatic recognition and mapping of rainfall induced shallow landslides using optical satellite images. – *Remote Sensing of Environment* 115, 7, 1743–1757.
- NETELER, M. & H. MITASOVA (2008<sup>3</sup>): Open source GIS: A GRASS GIS approach. New York: Springer.
- NEUBERT, M., HEROLD, G. & G. MEINEL (2008): Assessing image segmentation quality - concepts, methods and application. In: BLASCHKE, T., LANG, S. & G. HAY (eds.): Object-based image analysis: Spatial concepts for knowledge-driven remote sensing applications. Lecture notes in geoinformation and cartography. Berlin: Springer, 769–784.
- NPOCMAKA (2015): sendKeys.bat.  
<https://github.com/npocmaka/batch.scripts/blob/master/hybrids/jscript/sendKeys.bat>  
 (last access: 15-01-2017).
- OLAYA, V. (2009): Basic Land-Surface Parameters. In: HENGL, T. & H.I. REUTER (eds.): Geomorphometry: Concepts, software, applications. Developments in Soil Science 33. Amsterdam: Elsevier, 141–169.
- O'NEIL-DUNNE, J. (n.y.): LiDAR point cloud, Central Park, New York City: A high resolution image of the parent “point cloud” data.

- <[http://vcgi.vermont.gov/sites/vcgi/files/aboutvcgi/lidar/LiDAR\\_ptcloud\\_CentralPark\\_NYC.jpg](http://vcgi.vermont.gov/sites/vcgi/files/aboutvcgi/lidar/LiDAR_ptcloud_CentralPark_NYC.jpg)> (last access: 16-04-2017).
- PAU, G., FUCHS, F., SKLYAR, O., BOUTROS, M. & W. HUBER (2010): EBImage - an R package for image processing with applications to cellular phenotypes. – *Bioinformatics* 26, 7, 979–981.
- PEBESMA, E. (2017): sf: Simple Features for R. R package version 0.3-4.  
<https://CRAN.R-project.org/package=sf> (last access: 29-03-2017).
- PEBESMA, E.J. & R.S. BIVAND (2005): Classes and methods for spatial data in R.  
<http://cran.r-project.org/doc/Rnews/> (last access: 29-03-2017).
- PETSCHKO, H., BELL, R. & T. GLADE (2016): Effectiveness of visually analyzing LiDAR DTM derivatives for earth and debris slide inventory mapping for statistical susceptibility modeling. – *Landslides* 13, 5, 857–872.
- PIKE, R.J. (2000): Geomorphometry - diversity in quantitative surface analysis. – *Progress in Physical Geography* 24, 1, 1–20.
- PIKE, R.J., EVANS, I.S. & T. HENGL (2009): Geomorphometry: A Brief Guide. – *Developments in Soil Science* 33, 3–30.
- R CORE TEAM (2016): R: A language and environment for statistical computing.  
<https://www.R-project.org/> (last access: 05-01-2017).
- REVOLUTION ANALYTICS & S. WESTON (2015): foreach: Provides Foreach Looping Construct for R. R package version 1.4.3.  
<https://CRAN.R-project.org/package=foreach> (last access: 09-04-2017).
- RIEGL (2010): Data Sheet: LMS-Q560.  
[http://www.riegl.com/uploads/ttx\\_pxpriegldownloads/10\\_DataSheet\\_Q560\\_20-09-2010\\_01.pdf](http://www.riegl.com/uploads/ttx_pxpriegldownloads/10_DataSheet_Q560_20-09-2010_01.pdf) (last access: 18-12-2016).
- RIEGL (2012): Data Sheet: LMS-Q680i.  
[http://www.riegl.com/uploads/ttx\\_pxpriegldownloads/10\\_DataSheet\\_LMS-Q680i\\_28-09-2012\\_01.pdf](http://www.riegl.com/uploads/ttx_pxpriegldownloads/10_DataSheet_LMS-Q680i_28-09-2012_01.pdf) (last access: 18-12-2016).
- RILEY, S.J., DEGLORIA, S.D. & R. ELLIOT (1999): A terrain ruggedness index that quantifies topographic heterogeneity. – *Intermountain Journal of Sciences* 5, 1-4, 23–27.
- SANTOS, J.A.D., FALCAO, A.X., TORRES, R.d.S., GOSELIN, P.-H. & S. PHILIPP-FOLIGUE (2013): Semi-automatic Classification of Remote Sensing Images. Université de Cergy Pontoise. Cergy-Pontoise Cedex, France.

- SAPPINGTON, J.M., LONGSHORE, K.M. & D.B. THOMPSON (2007): Quantifying Landscape ruggedness for Animal Habitat Analysis: A Case Study Using Bighorn Sheep in the Mojave Desert. – *The Journal of Wildlife Management* 71, 5, 1419–1426.
- SCHULZ, W.H. (2004): Landslides mapped using LIDAR imagery, Seattle, Washington. – US Geological Survey Open-File Report 1396, 11.
- SCHULZ, W.H. (2007): Landslide susceptibility revealed by LIDAR imagery and historical records, Seattle, Washington. – *Engineering Geology* 89, 1-2, 67–87.
- SCHWARZ, G. (1978): Estimating the Dimension of a Model. – *The Annals of Statistics* 2, 6, 461–464.
- SHANNON, C.E. (1949): Communication in the Presence of Noise. – *Proceedings of Institute of Radio Engineers* 37, 1, 10–21.
- STABLER, B. (2013): shapefiles: Read and Write ESRI Shapefiles. R package version 0.7. <<https://CRAN.R-project.org/package=shapefiles>> (last access: 29-03-2017).
- STEGER, S., BRENNING, A., BELL, R. & T. GLADE (2017): The influence of systematically incomplete shallow landslide inventories on statistical susceptibility models and suggestions for improvements. – *Landslides* 2, 2, 1–15.
- STEHMAN, S.V. & R.L. CZAPLEWSKI (1998): Design and Analysis for Thematic Map Accuracy Assessment. – *Remote Sensing of Environment* 64, 3, 331–344.
- STEINIGER, S. & E. BOCHER (2009): An overview on current free and open source desktop GIS developments. – *International Journal of Geographical Information Science* 23, 10, 1345–1370.
- STUMPF, A. & N. KERLE (2011): Object-oriented mapping of landslides using Random Forests. – *Remote Sensing of Environment* 115, 10, 2564–2577.
- TARBOTON, D.G. (1997): A new method for the determination of flow directions and upslope areas in grid digital elevation models. – *Water Resources Research* 33, 2, 309–319.
- TARBOTON, D.G., DASH, P. & N. SAZIB (2015): TauDEM 5.3: Guide to using the TauDEM command line functions. <<http://hydrology.usu.edu/taudem/taudem5/TauDEM53CommandLineGuide.pdf>> (last access: 14-01-2017).
- TAROLLI, P. (2014): High-resolution topography for understanding Earth surface processes: Opportunities and challenges. – *Geomorphology* 216, 295–312.

- TAROLLI, P., SOFIA, G. & G. DALLA FONTANA (2012): Geomorphic features extraction from high-resolution topography: landslide crowns and bank erosion. – *Natural Hazards* 61, 1, 65–83.
- TRIMBLE DOCUMENTATION (2014): eCognition Developer 9.0: Reference Book. Munich.
- VAN DEN EECKHAUT, M., KERLE, N., POESEN, J. & J. HERVÁS (2012): Object-oriented identification of forested landslides with derivatives of single pulse LiDAR data. – *Geomorphology* 173-174, 30–42.
- VAN DEN EECKHAUT, M., POESEN, J., VERSTRAETEN, G., VANACKER, V., NYSSEN, J., MOEYERSONS, J., VAN BEEK, L.P.H. & L. VANDEKERCKHOVE (2007): Use of LIDAR-derived images for mapping old landslides under forest. – *Earth Surface Processes and Landforms* 32, 5, 754–769.
- VAN DER WERFF, H. & F.D. VAN DER MEER (2008): Shape-based classification of spectrally identical objects. – *ISPRS Journal of Photogrammetry and Remote Sensing* 63, 2, 251–258.
- VARNES, D.J. (1978): Slope Movement Types and Processes. – *Transportation Research Board Special Report*, 176, 11–33.
- VENABLES, W.N., SMITH, D.M. & R CORE TEAM (2016): An Introduction to R. Notes on R: A Programming Environment for Data Analysis and Graphics Version 3.3.2.  
[<https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf>](https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf) (last access: 31-10-2016).
- WEHR, A. & U. LOHR (1999): Airborne laser scanning—an introduction and overview. – *ISPRS Journal of Photogrammetry and Remote Sensing* 54, 2-3, 68–82.
- WILSON, J.P. & M.P. BISHOP (2013): Geomorphometry. In: SHRODER, J.F. (ed.): *Treatise on Geomorphology* 3. Amsterdam: Elsevier, 162–186.
- WILSON, J.P. & J.C. GALLANT (2000a): Digital Terrain Analysis. In: WILSON, J.P. & J.C. GALLANT (eds.): *Terrain analysis: Principles and applications*. New York: Wiley, 1–28.
- WILSON, J.P. & J.C. GALLANT (2000b): Secondary Topographic Attributes. In: WILSON, J.P. & J.C. GALLANT (eds.): *Terrain analysis: Principles and applications*. New York: Wiley, 87–132.
- WOOD, J.: The geomorphological characterisation of Digital Elevation Models. Dissertation. University of Leicester. Department of Geography. Leicester.  
[<https://ira.le.ac.uk/bitstream/2381/34503/1/U077269.pdf>](https://ira.le.ac.uk/bitstream/2381/34503/1/U077269.pdf) (last access: 20-02-2017).
- WOOD, J. (2009): Overview of Software Packages Used in Geomorphometry. – *Developments in Soil Science* 33, 257–267.

- YI, W. & S. MARSHALL (2000): Principal component analysis in application to object orientation. – *Geo-spatial Information Science* 3, 3, 76–78.
- YOËLI, P. (1965): Analytische Schattierung: Ein kartographischer Entwurf. – *Kartographische Nachrichten* 15, 5, 141–148.
- YOKOYAMA, R., SHIRASAWA, M. & R.J. PIKE (2002): Visualizing topography by openness: a new application of image processing to digital elevation models. – *Photogrammetric Engineering and Remote Sensing* 68, 3, 257–265.
- ZAKŠEK, K., OŠTIR, K. & Ž. KOKALJ (2011): Sky-View Factor as a Relief Visualization Technique. – *Remote Sensing* 3, 12, 398–415.
- ZALKA, T. (2016): Bitte um Fragenbeantwortung für eine Diplomarbeit zum Thema ALS-DHM und Rutschungen. e-mail correspondence with the federal state government of Burgenland (02-08-2016).
- ZEVENBERGEN, L.W. & C.R. THORNE (1987): Quantitative Analysis of Land Surface Topography. – *Earth Surface Processes and Landforms* 12, 1, 47–56.
- ZHOU, Q., LEES, B. & G.A. TANG (2008): Advances in Digital Terrain Analysis. Lecture notes in geoinformation and cartography. Berlin: Springer.
- ZUNIC, J. & P.L. ROSIN (2004): A new convexity measure for polygons. – *IEEE transactions on pattern analysis and machine intelligence* 26, 7, 923–934.
- ZVOLEFF, A. (2016): glcm: Calculate Textures from Grey-Level Co-Occurrence Matrices (GLCMs). R package version 1.6.1.  
<<https://CRAN.R-project.org/package=glcm>> (last access: 09-04-2017).

## Appendix

### A. Data of Segmentation Levels

Table A-1: Overview about outputs and classes used in segmentation levels (OWN REPRESENTATION).

output	class	description	number of objects
<b>segmentation level 1</b>			
curv_max15_IQR_buf (.sdat)		selected and buffered cells from maximum curvature grid	
L1_seg (.shp)		first segmentation	661,210
L1_seg_sel (.shp)			241,007
111		possible scarp candidates (L1)	8,730
44		intersection of sky-view factor ( $< 0.7432842$ ) and openness ( $< -2.5721536$ )	1,666
55		maximum curvature ( $\geq 0.00411455$ )	1,261
66		buffered flow accumulation depending on thresholds:	
	10,000	- 1 m	
	30,000	- 3 m	
	70,000	- 7 m	
	200,000	- 10 m	
	800,000	- 15 m	
	1,250,000	- 20 m	6,515
L1_final (.shp)	11	possible scarp candidates, refined (L1)	3,220
L1_final (.sdat)		rasterized output of <i>L1_final</i>	3,220
L1_final_buf (.sdat)		buffered (5 m) <i>L1_final</i>	
L1_final_pts (.shp)		one point of the surface of every object of <i>L1_final</i>	3,220
<b>segmentation level 2</b>			
L2_seg (.shp)		second segmentation	292,404
bounding box flow		scale.factor = c(1.4, 0.4) k.centroid = 10 centroid = FALSE set.centroid = "direction" scale.side = "small"	
bounding box inverse flow		scale.factor = 0.6 k.centroid = 4 centroid = FALSE set.centroid = "direction" scale.side = "small"	
L2_seg_sel (.shp)			25,269
11		possible scarp candidates (L1)	3,220
22		possible flanks	359
77		possible crowns	1,168
L2_flank (.shp)		possible flanks, refined (L2)	331
L2_flank (.sdat)		rasterized output of <i>L2_flank</i>	331
L2_flank_pts (.shp)		one point of the surface of every object of <i>L2_flank</i>	331

<b>output</b>	<b>class</b>	<b>description</b>	<b>number of objects</b>
L2_crown (.shp)		possible crowns, refined (L2)	794
L2_crown (.sdat)		rasterized output of <i>L2_crown</i>	794
L2_crown_pts (.shp)		one point of the surface of every object of <i>L2_crown</i>	794
L2_final (.sdat)		raster containing possible crowns, scarps and flanks	4,345
L2_final_buf (.sdat)		buffered (25 m) <i>L2_final</i>	
<b>segmentation level 3</b>			
L3_seg (.shp)		third segmentation	65,229
L3_seg_sel (.shp)			23,170
	landslides	final landslide objects	398
	1	final landslide scarps	640
	2	final landslide flanks	116
	3	final landslide bodies	2,243
	7	final landslide crowns	304
	30	potential landslide bodies	1,736
	130	merged objects of classes 1 and 30	2,376
	88	normalized height ( $\leq$ 0.08239538)	2,195
	89	normalized height ( $<$ 0.19024493)	5,836
	99	length-width-ratio ( $\geq$ 38.942666)	1,102
bounding box flow conv< thrs		scale.factor = c(1.4, 1.2) k.centroid = 3 centroid = FALSE set.centroid = "inverse" scale.side = "long"	
bounding box flow conv $\geq$ thrs		scale.factor = c(1.8, 1.2) k = 2 centroid = TRUE scale.side = "long"	
bounding box inv flow conv< thrs		scale.factor = 0.3 k.centroid = 2 centroid = FALSE set.centroid = "direction" scale.side = "small"	
bounding box inv flow conv $\geq$ thrs		scale.factor = 0.3 k = 4 centroid = FALSE set.centroid = "direction" scale.side = "small"	

## B. Results of Objective Function

Table B-1: Results of objective function in segmentation level 1 (OWN REPRESENTATION).

scale parameter	intrasegment variance	normalized intrasegment variance	Moran's I	normalized Moran's I	objective function
0.25	2.673549705	1	0.8626029	0	1
0.5	3.068078806	0.95884578	0.8100634	0.130362835	1.089208615
4	3.694469302	0.893505574	0.73680865	0.312125093	1.205630667
1	3.994638821	0.862194215	0.72698585	0.336497777	1.198691991
1.25	4.799212547	0.778267315	0.68050895	0.451817898	1.230085213
1.5	5.614256415	0.69324825	0.66547329	0.489124902	1.182373152
1.75	6.130659261	0.639381105	0.65589519	0.512890425	1.15227153
2	6.757552757	0.57398843	0.63961519	0.55328493	1.12727336
2.25	7.176141247	0.530324521	0.62846417	0.580953224	1.111277745
2.5	7.646335212	0.481277528	0.61652879	0.610567726	1.091845254
2.75	8.071339752	0.436944346	0.59912727	0.653744984	1.090689329
3	8.438563771	0.39863838	0.58436153	0.690382255	1.089020635
3.25	8.909833623	0.349479159	0.57237968	0.720112046	1.069591204
3.5	9.253702968	0.313609372	0.56383196	0.741320935	1.054930307
3.75	9.571011613	0.280510191	0.55348385	0.766997036	1.047507227
4	9.889091542	0.247330556	0.53661479	0.808853141	1.056183698
4.25	10.18438418	0.216527916	0.5257745	0.835750456	1.052278372
4.5	10.48778744	0.184879237	0.51869039	0.853327794	1.038207031
4.75	10.84958767	0.147139039	0.50167377	0.895550014	1.042689053
5	11.13727882	0.117129326	0.49970552	0.900433715	1.017563041
5.25	11.37880574	0.091935108	0.48549611	0.935690591	1.027625699
5.5	11.74613151	0.053618529	0.47637133	0.958331335	1.011949864
5.75	12.0885033	0.017904955	0.47254501	0.967825336	0.985730291
6	12.26015097	0	0.45957781	1	1

plateau function value: 1.161.

**Table B-2: Results of objective function in segmentation level 2 (own representation).**

scale parameter	intrasegment variance	normalized intrasegment variance	Moran's I	normalized Moran's I	objective function
0.25	0.023241317	1	0.849539507	0	1
0.5	0.027000062	0.953396954	0.786172618	0.198214188	1.151611142
4	0.032556081	0.884510281	0.713245601	0.426332841	1.310843122
1	0.036734281	0.832706594	0.680842129	0.527692209	1.360398803
1.25	0.04126942	0.77647737	0.647842248	0.63091717	1.40739454
1.5	0.048374113	0.688389371	0.600123625	0.780182948	1.468572319
1.75	0.052234561	0.640525351	0.59486384	0.796635769	1.437161119
2	0.058549395	0.562230463	0.571969252	0.868250965	1.430481429
2.25	0.063632881	0.499202522	0.567711903	0.881568123	1.380770645
2.5	0.070131265	0.418631891	0.552528824	0.929061408	1.347693299
2.75	0.075459447	0.352570075	0.547905818	0.943522359	1.296092434
3	0.082633297	0.26362462	0.540193365	0.967647221	1.231271841
3.25	0.088197986	0.194630461	0.536467446	0.979302046	1.173932507
3.5	0.094425674	0.117416051	0.531024987	0.996326278	1.113742329
3.75	0.098868049	0.062336972	0.529850539	1	1.062336972
4	0.103895806	0	0.53615163	0.980289933	0.980289933

plateau function value: 1.30770008238794.

**Table B-3: Results of objective function in segmentation level 3 (OWN REPRESENTATION).**

scale parameter	intrasegment variance	normalized intrasegment variance	Moran's I	normalized Moran's I	objective function
2	6.757552757	1	0.63961519	0	1
2.25	7.176141247	0.936839987	0.628464173	0.053966256	0.990806243
2.5	7.646335212	0.865893324	0.616528786	0.111728539	0.977621863
2.75	8.071339752	0.801765205	0.599127268	0.195944607	0.997709812
3	8.438491966	0.746366313	0.58440457	0.267196307	1.013562621
3.25	8.90927991	0.675330025	0.572445983	0.325070868	1.000400893
3.5	9.253930143	0.623326414	0.563908452	0.366388946	0.989715361
3.75	9.568304906	0.57589101	0.553474246	0.416886134	0.992777145
4	9.888306382	0.527606603	0.536558071	0.498753349	1.026359952
4.25	10.18159456	0.483352913	0.525668314	0.551455218	1.034808132
4.5	10.48834075	0.437068569	0.518742829	0.584971666	1.022040235
4.75	10.85347361	0.381974374	0.501695397	0.667474108	1.049448482
5	11.13645915	0.33927523	0.499709637	0.677084357	1.016359588
5.25	11.37909524	0.302664337	0.485490231	0.745900333	1.04856467
5.5	11.74561147	0.247361407	0.476220605	0.790761446	1.038122853
5.75	12.08934271	0.195496461	0.472538127	0.808583096	1.004079557
6	12.26442622	0.169078448	0.459782474	0.870315122	1.03939357
6.25	12.49590161	0.13415157	0.447633909	0.929109098	1.063260668
6.5	12.76445833	0.093629563	0.449579087	0.91969525	1.013324813
6.75	12.99324726	0.059108042	0.479091636	0.776866853	0.835974895
7	13.3849816	0	0.432985762	1	1

plateau function value: 1.01778128371373.