TECHNISCHE UNIVERSITÄT MÜNCHEN
Institut für Photogrammetrie und Kartographie
Fachgebiet Photogrammetrie und Fernerkundung

Car detection in low frame-rate aerial imagery of dense urban areas

Sebastian Türmer

Dissertation

2014

TECHNISCHE UNIVERSITÄT MÜNCHEN
Institut für Photogrammetrie und Kartographie
Fachgebiet Photogrammetrie und Fernerkundung

Car detection in low frame-rate aerial imagery of dense urban areas

Sebastian Türmer

Vollständiger Abdruck der von der Ingenieurfakultät Bau Geo Umwelt der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr. phil. nat. Urs Hugentobler
Prüfer der Dissertation:	1. Univ.-Prof. Dr.-Ing. Uwe Stilla
	2. Hon.-Prof. Dr.-Ing. Peter Reinartz Universität Osnabrück
	3. Univ.-Prof. Dr. rer. nat. Ralf Reulke Humboldt-Universität zu Berlin

Die Dissertation wurde am 01.10.2013 bei der Technischen Universität München eingereicht und durch die Ingenieurfakultät Bau Geo Umwelt am 11.04.2014 angenommen.

Abstract

Knowledge about quantity and position of moving and stationary vehicles is essential for traffic management and planning. This information can be used, for instance, for security of mass events or to support rescue crews in disaster situations. In order to get this information, large areas have to be examined quickly and completely. Very suitable for this task are airborne optical sensors. However, a reliable automatic method to locate vehicles in aerial images is necessary.

In the present work a method for automatic extraction of vehicles in urban areas is presented. The work mainly covers three key fields of car detection. The first is related to the extraction of ground areas. On the assumption that trafficable areas are often ground areas in densely populated cities, disparity maps are calculated using the semi-global matching algorithm (SGM). Subsequently, a threshold is automatically determined to separate ground from non-ground regions (Minimum Error Thresholding). The second field concerns the introduction of a object-based method for extracting car candidates. In order to do this, the image is smoothed using the mean curvature flow, and a region-growing algorithm is then applied. The regions obtained are considered autonomous regions and are filtered multiple times with regard to their geometric properties. The third field is the examination of the remaining candidate regions by a classifier based on gradients (HOG features), which is trained by a machine learning algorithm (AdaBoost). However, the classifier is trained using only a few training samples. The goal is to minimize the manual effort and to provide a high degree of generalization.

Thus, a strategy is presented which combines object-based and gradient-based techniques. The strategy is tested with five urban images from the 3K+ camera system and the UltraCam Eagle camera system, with 13 cm and 20 cm GSD, respectively. Through the use of disparity maps, it is shown that the car detection quality in densely populated inner-city areas can be enhanced. Objects on the top of buildings are now accurately excluded from the detection process. Furthermore, the car detection approach presented is able to detect cars in different datasets without adjustment of parameter settings (different sensors and different resolution). The results of detection show that a completeness of 80% leads to a correctness of 65% to 95%.

Kurzfassung

Das Wissen von Anzahl und Position bewegter und stehender Fahrzeuge ist wichtig für Verkehrsmanagement und -planung. Aufgrund dieser Information kann beispielsweise die Sicherheit von Massenveranstaltungen erhöht werden oder Rettungskräfte können im Katastrophenfall unterstützt werden. Zur Gewinnung dieser Information sind die interessierenden Gebiete aktuell und flächig aufzunehmen. Hierfür eignen sich besonders flugzeuggestützte, optische Kamerasysteme. Allerdings ist zur automatischen Auswertung dieser Luftbilder ein zuverlässiges Verfahren notwendig, um die Fahrzeuge zu detektieren.

In der vorliegenden Arbeit wird ein Verfahren zur automatischen Extraktion von Fahrzeugen in städtischem Gebiet vorgestellt. Das Verfahren kombiniert eine objektbasierte mit einer gradientenbasierten Strategie und ist in drei Hauptbereiche unterteilt. Der erste Bereich behandelt die Extraktion von Bodenflächen. Unter der Annahme, dass für Fahrzeuge befahrbare Flächen in dicht besiedelten Städten meistens Bodenflächen sind, werden Höhenbilder mit dem Semi-global Matching Algorithmus (SGM) berechnet. Danach wird automatisch ein Grenzwert bestimmt, um Bodenflächen von Nicht-Bodenflächen zu trennen (Minimum Error Thresholding). Im zweiten Bereich wird ein objektbasiertes Verfahren eingeführt, um Fahrzeugkandidaten zu bestimmen. Hier wird zunächst die zeitliche Veränderung des Bilds aufgrund des Krümmungsflusses genutzt, um das Eingabebild zu glätten. Im nächsten Schritt wird ein Regionenwachstumsverfahren angewendet. Die erhaltenen Regionen werden als selbständige Objekte betrachtet und nach ihren geometrischen Eigenschaften mehrfach gefiltert. Der dritte Bereich beschreibt die Untersuchung der verbleibenden Fahrzeugkandidaten mit einem gradientenbasierten Klassifikator (HOG-Merkmale), welcher mit einem maschinellen Lernverfahren (AdaBoost) trainiert ist. Dieser Klassifikator ist jedoch nur mit wenigen Beispielen und Iterationsschritten trainiert.

Das Verfahren wird mit fünf innerstädtischen Luftbildern des 3K+ Kamerasystems (13 cm Bodenpixelgröße) und des UltraCam Eagle Kamerasystems (20 cm Bodenpixelgröße) getestet. Aufgrund der Verwendung von Höhenbildern kann die Qualität der Fahrzeugerkennung in dicht besiedelten innerstädtischen Gebieten erhöht werden. Objekte auf dem Dach von Gebäuden werden nun vom Detektionsprozess ausgeschlossen. Weiterhin ist das Verfahren fähig, ohne die Anpassung der Parameter, Fahrzeuge in unterschiedlichem Datenmaterial (verschiedene Sensoren mit unterschiedlicher Auflösung) zu erkennen. Die Ergebnisse der Detektion zeigen, dass bei einer Vollständigkeit von 80% eine Korrektheit zwischen 65% und 95% erreicht wird.

Contents

List of Figures

List of Tables

List of Abbreviations

ATKIS	authoritative topographic-cartographic information system
BKG	federal agency for cartography and geodesy
BRF	boosted random field
CCD	charge-coupled device
CCH	color co-occurrence histogram
C-HOG	circular histogram of oriented gradients
CHOG	compressed histogram of oriented gradients
CMOS	complementary metal-oxide-semiconductor
CPM	color probability map
CRF	conditional random field
DCM	directional chamfer matching
DEM	digital elevation model
DPM	deformable parts model
DSM	digital surface model
DTM	digital terrain model
EOH	edge orientation histograms
ESF	edge spread function
FC	feature context
FCD	floating car data
FEM	finite element method
FMC	forward-motion compensation
FPS	frames per second
GIS	geographic information system
GNSS	global navigation satellite system
GPS	global positioning system
GPU	graphics processing unit
GSD	ground sampling distance
HDHR	histogram distance on haar region
HOG	histogram of oriented gradients
HSV	hue saturation value
ICA	independent component analysis
IMU	inertial measurement unit
IR	infrared
ISO	international organization for standardization
KNN	k-nearest neighbors
LBP	local binary patterns
LiDAR	light detection and ranging

LSF	line spread function
MAD	multivariate alteration detection
NMF	non-negative matrix factorization
OSM	openstreetmap
PCA	principal component analysis
PLS	partial least squares
POP	pairs of pixels
RANSAC	random sample consensus
RF	random forest
RGB	red green blue
R-HOG	rectangular histogram of oriented gradients
RPAS	remotely piloted aircraft systems
SAR	synthetic aperture radar
SC	sparse code
SGM	semi-global matching
SIFT	scale-invariant feature transform
SIMD	single instruction multiple data
SMD	salient feature match distribution matrix
SPM	spatial pyramid matching
SRTM	shuttle radar topography mission
SURF	speeded up robust features
SVM	support vector machine
TIR	thermal infrared
UAV	unmanned aerial vehicle
UTM	universal transverse mercator

1 Introduction

1.1 Problem and motivation

"You're not stuck *in* the jam, you *are* the jam". This graffiti written on a wall next to a busy street reminds drivers that they are part of the traffic problem, rather than just innocent victims. Considering the fact that the amount of vehicle miles traveled has increased by nearly 100 percent over the last two decades [U.S. Department of Transportation, 2008], it is not surprising that the average hours of congestion each day have increased as well [Taylor, 2010]. Nowadays it is common knowledge that being caught in a traffic jam is not only annoying but also has a negative impact on the economy as well as the environment [Schrank et al., 2011]. The 2.9 billion gallons of petrol wasted in U.S. traffic jams in 2005 could fuel U.S. daily transportation needs for nearly a week (6.1 days) [U.S. Department of Transportation, 2005]. In order to prevent worse future scenarios, demanding solutions and further progress in research are required [Stantchev & Whiteing, 2010; Winder et al., 2010; Banister et al., 2010; Stilla et al., 2005, 2009].

However, congestion is not the only important topic. Other car-related topics like logistic and urban planing include parking space management [Huang & Wang, 2010] and parking behavior analysis [Nurul Habib et al., 2012]. Moreover, due to the increasing population in urban areas, resulting in additional traffic volume, especially in rapidly developing cities like Beijing (China) [Lv et al., 2011; Xiao et al., 2011] or Delhi (India) [Pucher et al., 2007], further problems arise such as air pollution, noise, energy use, traffic injuries and fatalities, congestion, parking shortages, and a lack of mobility for the poor. This poses questions to traffic planners who work on solutions which are often based on traffic data and models [Leonhardt, 2008; Hinsbergen, 2010]. Traffic models are also valuable in short-term situations like mass events or disasters [Pel et al., 2012].

Traffic data can be captured in various ways and positions. In order to face all aspects of traffic, the combination of several acquisition techniques delivers complementary information. A widely used low-priced solution is induction loops [Clark, 1983; Davidson & Valentine, 2001]. Induction loops are cable loops which are under the surface of roads and act as inductor. The inductance changes if a metallic object is in its range. They gather traffic data continuously, but only at isolated spots. In contrast to induction loops, stationary video cameras [Shillman & Schatz, 2011; Matsur, 2011; Bischof et al., 2010] allow us to exploit geometric information and unique identification, but also just locally. They are often installed on highly frequented streets. In addition to stationary sensors, the floating car principle [Albrecht et al., 1995] gives information about the traffic flow. Floating car data (FCD) are generated by utilizing the location of certain cars which are part of the current traffic pattern. The location and the velocity of the car is often

determined by GPS and mobile phone tracking [Busch et al., 2004]. Companies that provide such services are, for instance, TomTom [TomTom, 2009] or Google [Google, 2009]. However, only road users who agree to share their current position are monitored. Hence, this method does not allow us to collect data in regard to quantities. In addition, vehicle types and parked cars are not considered.

Generally, remote sensing enables us to gather geo-information from a distance. A collection about research on airborne and spaceborne traffic monitoring is given in Hinz et al. [2006]. Spaceborne sensors are especially useful for mapping very large areas. Moreover, it is also shown that cars can be automatically extracted from satellite images [Sharma et al., 2006; Jin & Davis, 2007; Larsen et al., 2009; Eikvil et al., 2009; Leitloff et al., 2010; Leitloff, 2011; Salehi et al., 2012; Meng & Kerekes, 2012]. Unfortunately, they have drawbacks due to their limited flexibility. Many satellites operate in a sun-synchron mode which restricts them to certain periods of time and thus a low repetition rate. Additionally, they often have a low GSD (usually larger than 50 cm panchromatic). A more flexible option are airborne sensors operating on helicopters [Nejadasl et al., 2006], UAVs (unmanned aerial vehicle) [Breckon et al., 2008; Gleason et al., 2011] or airplanes.

Known airborne approaches deal with active sensors such as SAR and LiDAR or passive ones such as thermal infrared (TIR), hyperspectral, and other optical sensors in the visual domain. Traffic data acquisition with SAR [Palubinskas & Runge, 2007; Maksymiuk et al., 2012] has the major advantage of being independent from the weather. Due to progress in SAR sensors and data processing, leading edge data acquisition allows vehicle type classification [Brenner et al., 2012]. Also velocities can be derived by moving target indication [Ender et al., 2008; Cerutti-Maori et al., 2008; Baumgartner & Krieger, 2011]. While the interpretation of urban areas from SAR data is problematic due to the inherent side looking geometry [Stilla et al., 2004]. LiDAR allows nadir view in urban areas and can be used for car detection [Yao & Stilla, 2011] and as well for velocity estimation of vehicles [Yao et al., 2011, 2012]. However, LiDAR is based on monochromatic light and can not provide color information. Also typical for LiDAR is that every surface point is registered only once, in contrast to optical image sequences where multiple information is gathered of the same object. Image sequences do not only deliver multiple acquisition but also a denser sampling of the surface. Generally, the focus is on optical image sequences to which also IR cameras belong [Stilla & Michaelsen, 2002; Hinz & Stilla, 2006; Kirchhof & Stilla, 2006]. They provide a high frequent image acquisition and additionally supplemental information concerning the activity state of the vehicles. Warm parts (engine, body, etc.) appear as bright areas in the image which makes it possible to distinguish between stationary and parked cars [Yao et al., 2009]. Unfortunately, IR cameras only have a small pixel matrix and thus a low resolution. Similarly, hyperspectral sensors also provide a low resolution but they are often used for vehicle extraction [Manolakis et al., 2003; Casasent & Chen, 2003; Li et al., 2009]. Hyperspectral information can be used to exclude areas of vegetation or to determine shadow areas before the extraction process [Shimoni et al., 2011].

Sensors in the visual domain such as video cameras also have the ability to acquire high frame-rate image sequences which make it possible to observe the dynamics of traffic (Section 2.1). All in all, they have larger pixel matrices, but only offer lower resolution (in case of the same field of view) compared to single frame cameras. Cameras can be

distinguished between video cameras with a high frame rate (typically 24 to 30 FPS) and single frame cameras up to a few frames per second. However, the differences between these two categories are narrowing lately. Furthermore, professional aerial camera systems such as the UltraCam Eagle or the Quattro DigiCAM are not able to provide a frame rate higher than 1 Hz. This study focuses on exploiting image sequences from camera systems that allow us to capture high resolution images with 0.5 to 3 Hz. Thus the desired properties – high spatial resolution, large coverage, and multiple information of the same object – are fulfilled.

1.2 Objectives

The main objective of this dissertation is the development and the detailed analysis of a processing chain for car detection in aerial image sequences. Appropriate methods are restricted because in contrast to video data, the image acquisition rate is only low frequent (between 0.5 and 3 Hertz). The intention is to present a technique which detects cars in imagery of one and two decimeters GSD. The focus is not only on moving cars but also on parked cars. Furthermore, the position and orientation of the sensor in the aircraft is used which can be achieved by on-board GPS receivers and IMU instruments. Supplementary information utilized is derived from road databases.

A common problematic issue is the inaccuracy of road databases in urban areas. Often road databases are acquired by Global Navigation Satellite Systems (GNSS) [NAVTEQ, 2010; Zhou et al., 2013]. Roads can hardly be accurately recorded in areas with high buildings and urban canyons due to a lack of satellites from GNSS. Sometimes even road databases are not available due to frequently moving construction sites. Generally, these databases are mainly used for navigational tasks for which their accuracy is sufficient. However, in the case of car extraction they are usually used to extract roads or areas where cars are expected in order to limit the search area. Often, this application requires a more precise solution. Therefore, 3-dimensional information is exploited in order to support the overall car detection [Tuermer et al., 2013]. A information which can be derived from two subsequent images or in a different way.

Currently, many approaches for car detection use standard object detection methodology, in which detectors based on high-level features are trained with machine learning algorithms (see Section 2.2). Drawbacks of current methods can be the manual interaction during the training step and the missing robustness when the properties of the data change due to another sensor. Additionally, a top-performing detector must receive carefully selected training data and iterative back porting of false positives (e.g., online training [Grabner, 2008]). This back porting needs to be critically observed because a drifting of the detector must be avoided. This means that the detector is trained using certain false negative samples, it could omit some important positive detections as consequence. Consequently, a further goal is to develop a strategy with a simple parameter setting which is robust to changing resolution ranging from one to two decimeters, and the manual training effort should be as low as possible.

This dissertation focuses on car detection in aerial images of urban areas towards an elaborate extraction technique in the case of mass-events and catastrophes. These two scenarios fit the conditions where the benefits of airborne missions, like rapid availability and coverage of large regions, are exploited particularly useful [Kurz et al., 2012].

1.3 Outline

The following chapter 2 includes a literature review concerning vehicle detection in optical imagery and its special application for aerial optical imagery with low imaging frequency. After the introduction to the state of the art of car extraction techniques, the suggested car extraction strategy is shown in chapter 3. In chapter 4 the utilized airborne test data sets are described and the way of conducting the experiments is explained. Subsequently, the results of the experiments aiming to evaluate strategies related to car extraction are shown in chapter 5. Then results will be discussed regarding the method's drawbacks and potentials in chapter 6. In the last chapter it will be concluded with problems for car detection and ways to tackle them. Also potential developments for vehicle detection from aerial imagery in the near future with an expected higher resolution from UAVs are addressed.

2 Review of Related Literature

This chapter informs about previous research activities of vehicle detection in optical images. The first section presents methods that are based on optical imagery, in general, not necessarily related to remote sensing. The second section presents publications which are directly related to the present situation and its limitations in this dissertation.

2.1 Vehicle detection in optical images/videos

In order to put vehicle detection in low frame-rate aerial imagery (Section 2.2) into a comprehensive context, this section provides a short overview of relevant methods to detect vehicles in optical imagery. The first part is related to ground-based sensors (Section 2.1.1) and the second part to airborne sensors (Section 2.1.2). This grouping is done because cars seen from above look different compared to the typical side view.

2.1.1 Ground-based sensors

The following approaches use data from ground based sensors, many of them are based on video data. Nevertheless, ideas that were developed in that field have been sometimes brought to the remote sensing field as well. A further commonality of publications in the first part is the on-board or side view of cars.

On-board sensors – side view of cars

Methods aiming to detect cars from side view are very popular and have been carried out for several decades. A reason is that these images are widely available and the number of applications (keyword: driving safety systems) is huge. Often, publications in this field have introduced new ideas for object detection in general. Due to the vast number of publications only a few path-breaking ones can be mentioned in the following paragraph.

One of the early approaches [Dubuisson & Jain, 1995], here mentioned, extracts contours by using difference images, color segmentation and the Canny edge operator. The resulting contour is adapted by the snakes algorithm. However, the contour of a car seen from the side allows a better separation from other objects than the contour of a car which is seen from above. The reason is that the shape of a car which is seen from the side is more unique compared to other objects than the shape of a car seen from above. Cars seen from above have, with only a few exceptions, a rectangular shape. Regardless of the contour,

different features are used in the work of Schneiderman & Kanade [2000]. They use quantized wavelet coefficients in combination with AdaBoost. In the same year, Haar-like features showed their suitability for car detection, together with a support vector machine (SVM) [Papageorgiou & Poggio, 2000].

A framework for modeling the relationship between context and object properties based on the correlation between the statistics of low-level features is shown by Torralba [2003]. In a later study of boosted random fields (BRFs), Torralba et al. [2005] use the boosting method to learn the graph structure and local evidence of a conditional random field (CRF). CRFs are very useful to keep the information of the relation of certain segments. An application for aerial images could be the detection of cars which park in a row along the road; single cars parked elsewhere for example in a backyard are more challenging. With a similar intention a global feature is introduced by Murphy et al. [2006]. Steerable pyramids are used which pay attention to dominant textural features of the overall image, and to their coarse spatial layout. The basic method consists of several standard filter banks and the gentle AdaBoost algorithm.

The AdaBoost algorithm is used also by Negri et al. [2008]. They show a solution for car extraction using Haar-like and HOG features which are selected and weighted by the real AdaBoost algorithm. Further, Perrotton et al. [2009] use gentle AdaBoost and added additional features such as histogram distance on Haar region (HDHR), edge orientation histograms (EOH), HOG and Gabor filters. The idea is that new feature families should only be introduced if these features already used are not sufficient for classification. The same author [Perrotton et al., 2010] presents a work utilizing a soft cascade structure of the classifier. Stages of the cascade correspond to the partial sum of weak classifiers. In order to get a multi-view weak classifier, the selection of weak classifiers is carried out in a different way as done in the original work of Viola & Jones [2001]. Again Haar wavelets and different learning techniques (SVM, AdaBoost) are examined in the thesis of Zehnder [2009]. Furthermore, once again Haar-like features but online boosting are used in the work of Chang & Cho [2010]. A work which uses gentle AdaBoost tries to combine the detection and the segmentation process [Torrent et al., 2011].

A completely different strategy is pursued by Leibe et al. [2008]. In their work the information of features from different training samples is put together by using the center of similar features in the feature space. Resulting vectors are stored in a codebook (similar to the Bag-of-Words approach). The approach of Givoni et al. [2011] introduces also an interesting idea because videos and not static images are used in the training step. Afterwards optical flow, HOG features, and a Bag-of-words model are used for the training. Finally, the resulting classifier can be applied to static objects in single images as well.

Similarly, Wang & Lien [2008] take up the basic idea of the Bag-of-words method and use sub-regions of vehicles which are projected to eigenspace and independent basis space in order to generate a principal component analysis (PCA) weight vector and an independent component analysis (ICA) coefficient vector. Based on the joint probability of these vectors a likelihood estimation is carried out. Also shape features can be used [Lim et al., 2009] which are extracted at the location found by interest point operators. In addition, the detection has been assisted by extracting the lane region and a measurement

of symmetry. A review of vehicle detection methods where the camera is mounted on the vehicle up to the year 2006 can be found in Sun et al. [2006].

Stationary sensors – oblique view of cars

Data received by stationary video cameras, and thus showing an oblique view, is the basis of the following approaches.

One suggestion is the use of optical flow and 3D contours [Haag & Nagel, 1999]. Additionally, a 3D scene model, a lane model, an illumination model, and a camera model which is easily available due to the fixed camera position are incorporated. Unfortunately, optical flow is only applicable in the case of small changes thus high frequent video data better suit this approach.

Furthermore, a suitability evaluation of color histograms for vehicle detection can be found in Knauer et al. [2005]. Another work also based on color values uses a special color transform and generates a Bayesian classifier [Tsai et al., 2005]. Edge maps and coefficients of a wavelet transform are used to verify the detected candidates. In a similar manner wavelets are utilized by Salem & Meffert [2007]. However, they rely on a 3D wavelet based algorithm where time is the third dimension.

An adaptive background estimation technique plus histograms of gray values and edges from difference images is illustrated in the work of Zhou et al. [2007]. Also aiming to detect cars from oblique view, Roth et al. [2009] present a method relying on Haar-like features and online Boosting. Additionally, they generate separate classifiers for different image locations. Moreover, a work which proposes an adaptive threshold estimation for edges after applying the Sobel filter in order to cope with problems due to changing illumination conditions is presented by Laparmonpinyo & Chitsobhuk [2010]. In the end a benchmark schema has been made available by Kasturi et al. [2009]. Their base line algorithm for comparison to state of the art methods uses background subtraction plus a blob filtering.

2.1.2 Airborne sensors

Airborne sensors have been used in the second category where the popularity of UAVs has increased within the last few years. A great number of these approaches has been carried out on video data (high imaging frequency). Methods that work here are not necessarily transferable to the low frame rate case. For instance, popular methods like the optical flow cannot be applied when the time between the changes is too great (non-video data), because the new position of the moving pixel is too far away from its original position and cannot be identified again.

However, in the case of video data the use of optical flow and a statistical decision is possible [Nejadasl, 2005; Nejadasl et al., 2006; Nejadasl, 2010]. The same authors explored also a way for background calculation of gray value images [Nejadasl & Lindenbergh, 2011]. Pixels that exceed a certain value in the next frame are considered to belong to the foreground objects.

The idea of difference images and GIS road masks is used by Mirchandani et al. [2002]. The images are taken by a sensor mounted on a helicopter with GPS and IMU. Similarly, difference images are used in a further work [Cao et al., 2011a, 2012a]. Each frame is divided in layers where background and foreground objects are described by a Shi-Tomasi corner detector.

Difference images of the stabilized scene and a moving object model are also used to detect cars in thermal infrared images [Kirchhof & Stilla, 2006]. To distinguish moving cars from other objects, such as higher buildings, features like eccentricity and mass of the resulting elliptical blobs are used. As a constraint, a reasonable velocity of the cars is assumed to reject false positives.

Another way to determine relevant objects in the foreground is shown in the following works. These relevant areas are called salient locations at which HOG features are calculated, afterwards the matching is done by comparing them in the introduced salient feature match distribution matrix (SMD) [Khan et al., 2010]. The comparison of the features in the SMD is done based on their Euclidean distance. The salient locations are manually chosen.

Similarly, Cao et al. [2012b] also aim to extract salient locations first, therefore saliency maps are calculated as a kind of pre-processing. These maps consist of layers based on color, Gabor and motion features. The final classification is done by Haar-like features and AdaBoost. Another publication by Cao et al. [2011c] shows a strategy which generates several classifiers by discrete AdaBoost for certain parts of the vehicle. The output of all boosted classifiers is further classified using a SVM. The same authors present a way of calculating a feature similar to HOG with lower dimensionality [Cao et al., 2011b]. At the end the final classification is also done by a linear SVM.

Finally, Cheng et al. [2012] shows a way to identify background colors using a color histogram. Then advanced features based on the Harris corner detector and the Canny edge detector are calculated. Additionally, the result of a SVM which classifies color values after a color transformation is used as a feature. Finally, all features are passed to a dynamic Bayesian network for classification.

2.2 Airborne vehicle detection in low frame-rate optical image sequences

The automatic detection of vehicles from airborne optical sensors in single images or image sequences (up to 3 Hz) has been pursued by several researchers within the last few years. A graphical visualization of these publications can be seen in Figure 2.1. Often there are two major components of each approach, the utilized feature and the algorithm in order to classify the feature space. The categorization in this figure is according to the utilized features. The decision for that kind of classification has been chosen, because the impact concerning the detection quality is highly dependent on the descriptive elements. The following detailed description of the techniques is separated by headings which correspond to Figure 2.1.

Alternatively, in Figure 2.2, the publications are grouped according to the utilized classification strategy. However, the separation is sometimes more fuzzy compared to the grouping based on features (Figure 2.1). Some approaches utilize more than one algorithm which leads to ambiguities when a stereotypical grouping is aimed.

2.2.1 Single image

In this section all methods are based on the information of one image. The arrangement is according to the branch of single images in Figure 2.1.

Gradient-based

Contour Burlina et al. [1997] combines contours obtained by the Canny edge detector and votes obtained by the Hough transform. The generalized Hough transform of the image is calculated using the known shape and size of the sample car. If shape and size match to a car, a vote is created in the center of the hypothetical car. Finally, when the resulting values of the edge map and the value from the Hough transform exceed a certain threshold it is accepted as a car. The threshold is determined by a Bayesian strategy and a Neyman-Pearson strategy. It also shows first signs of online learning where parameters are re-adjusted during the detection procedure. Additionally, they add the feature of vehicle formations where periodic object configurations such as convoys on roads or vehicles in parking lots are used.

The Canny edge detector and the Hough transform have also been utilized in the approach of Moon et al. [2002] where the basic idea is the creation of a car model which consists of four edge detectors having the size and the shape of an average car. The candidate is only accepted when all four edges give a feasible feedback. The testing data shows vehicles in an average size of 7 by 17 pixels. Long shadows, for instance, from low illumination angles lead to false positive detections, and very oblique camera angles are also a source of errors.

Aiming to take advantage of the simplicity and the resulting low computational load, an improved version appeared some years later [Kozempel & Reulke, 2009]. In contrast to the previous approach, they created four special shaped edge filters to represent all edges of the car model. However, due to the simple model (rectangle) many false alarms (like vegetation pattern) have to be dealt with. An extension is shown by validating the previously received hypotheses [Kozempel, 2012]. For that task SURF features are utilized. The final classification is pursued using a SVM based on a radial basis function [Hausburg, 2010].

The technique of template matching is pursued by Pelapur et al. [2013]. An object is examine by calculating the distance of its edge map to template edge maps. The distance is calculated using the directional chamfer matching (DCM) method. Additionally, two different ways of calculating the initial edge maps have been compared with regard to their performance. Results showed that edge maps calculated by the multiscale Hessian-based line segment feature extraction method are superior to edge maps calculated by the

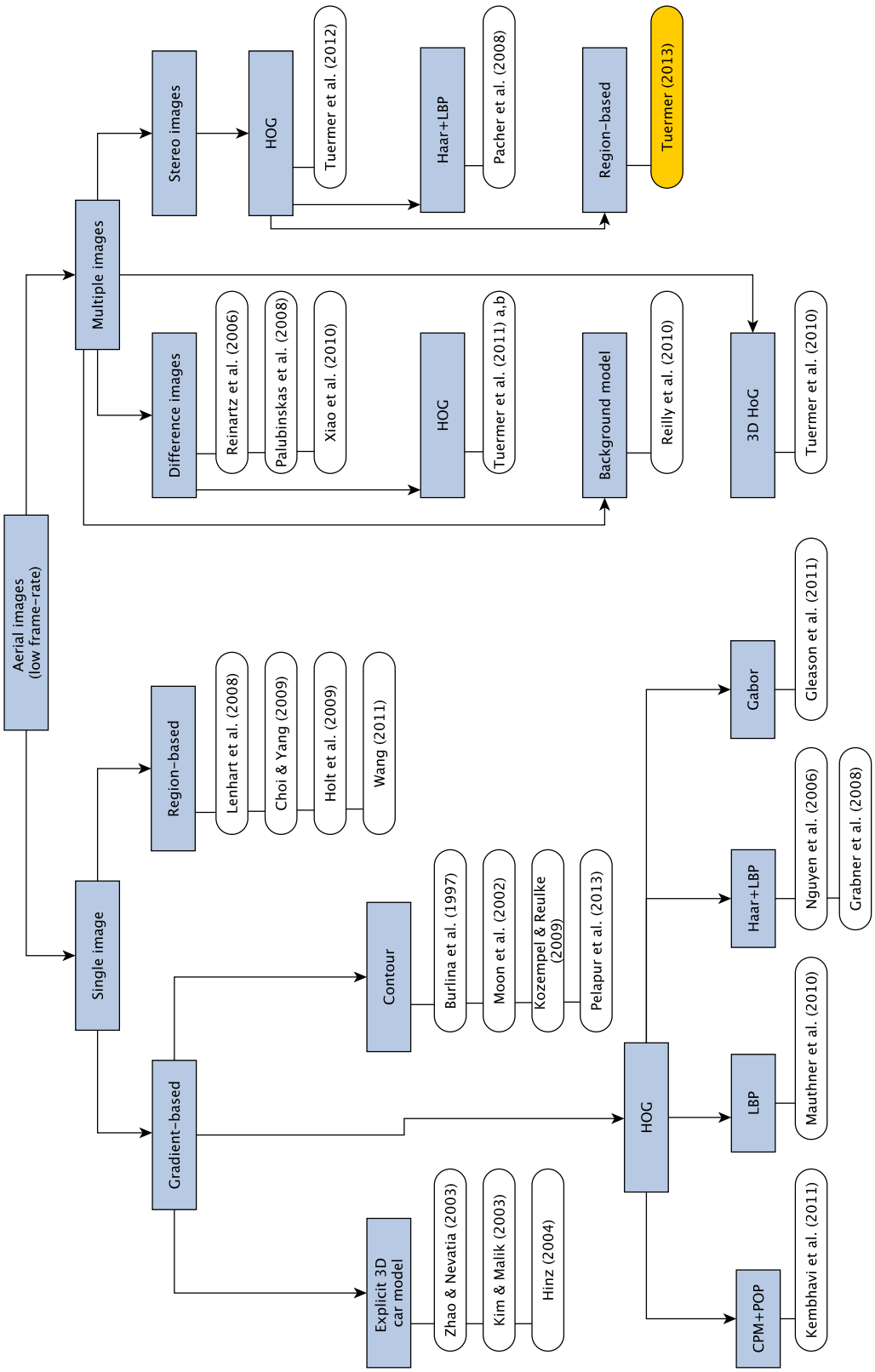


Figure 2.1: Overview of literature related to vehicle detection in low frame-rate aerial images. The publications are grouped according to the utilized features. The presented strategy is yellow.

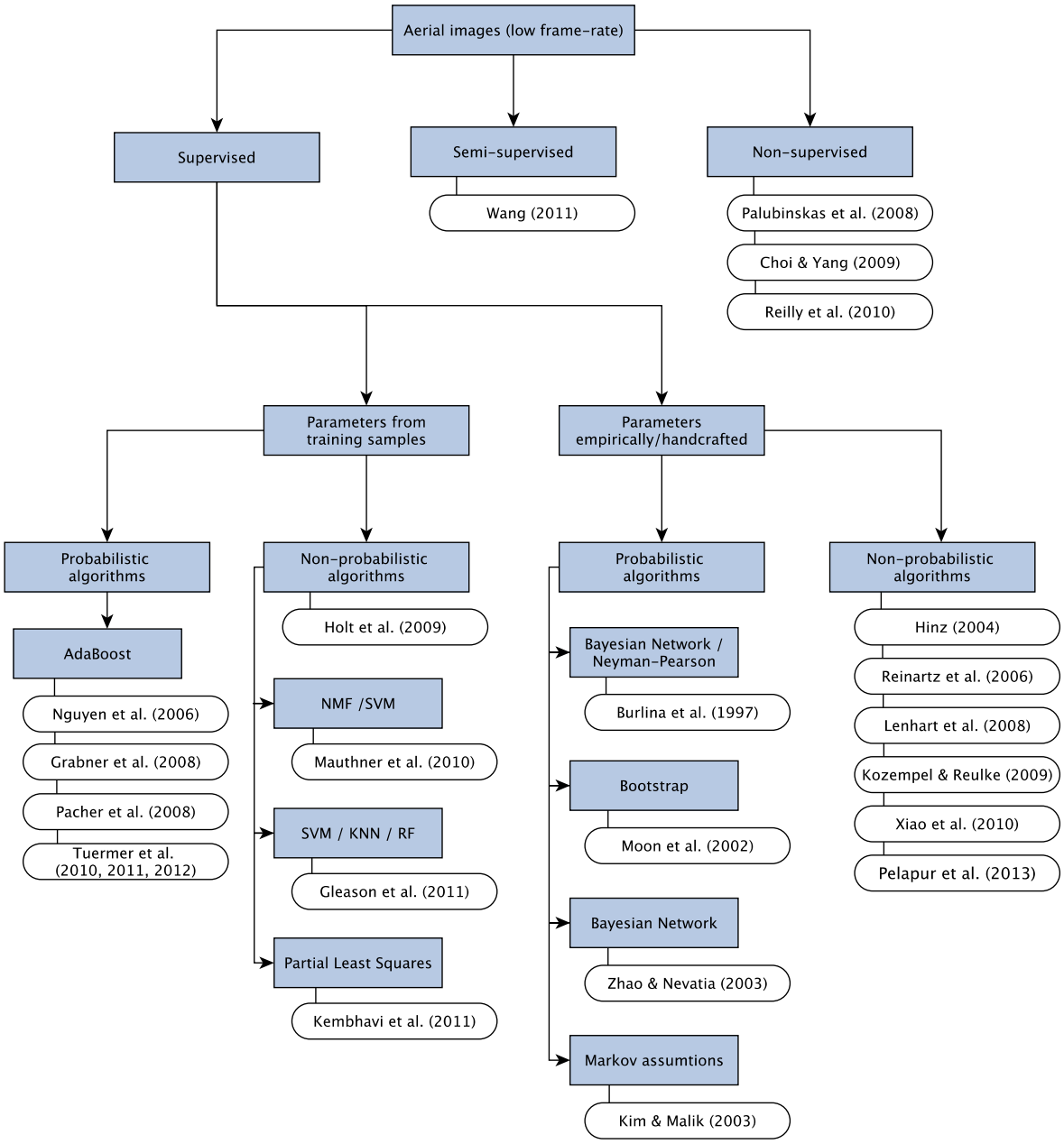


Figure 2.2: Overview of literature related to vehicle detection in low frame-rate aerial images. The publications are grouped according to the utilized classification strategy. The presented strategy is a combination of the probabilistic trunk, where the parameters are derived from training data, and from the non-probabilistic trunk, where the parameters are empirically determined.

Canny edge detector. Generally, the main focus is on determining the orientation of the vehicles. For this purpose a radon transformation is utilized.

Explicit 3D car model A more complex model is based on a wire frame consisting of features such as body boundary and windshield [Zhao & Nevatia, 2003]. The high impact of shadow, which is typically located on one side of the car is mentioned, as well as the intensity of the shadow is even suggested as an optional feature. The final decision is based on the probability and all features are passed to a Bayesian network with manually selected parameters. Directions of interest are determined by calculating a histogram of Canny edges. It is assumed that the main direction has the highest peak of the histogram. More false positive and false negative detections occur for dark cars as they have fewer salient features. Most false positives result from rectangular shapes of structures in buildings, tree foliage or road markings.

The 3D structure is relevant where line features are fitted to a car model [Kim & Malik, 2003]. In this case, the previous 2D line features are received by the Canny edge detector. Afterwards, a connected-component analysis is performed to group them. It is assumed that the rear line, front line, left and right line of the roof are always detected. The probability density function for every line is estimated from training samples. Finally, the complete system consisting of all line features is probabilistically evaluated and classified. The authors report difficulties due to distracting lines which are from tree shadows. Another issue occurs when parked cars have too little space between each other.

The method of Hinz [2004] enhances the car model idea by creating a local model of an average car describing the most prominent geometric and radiometric features. The intensity of the shadow is also incorporated, and the position of the sun is determined by internal and external image orientation parameters. Additionally, a global model is added for which vehicle queues are modeled as ribbons that exhibit the typical symmetry and spacing of vehicles. A disadvantage is the large number of necessary models which is tackled by introducing a tree-like model hierarchy. Problems occur due to weak contrast, specularities, occlusions and vehicle geometry which were not modeled by the explicit top-down procedure.

HOG One of the implicit methods [Nguyen et al., 2006; Grabner et al., 2008] makes use of the histograms of oriented gradients (HOG), Haar-like features and local binary patterns (LBP). These features are passed to an online Boosting algorithm to generate a strong classifier. The focus is on the online ability of the machine learning algorithm. An almost perfect detector can be obtained when reusing the false positives as negative training samples for the next training round, but this is a manual task. A smart approach trying to automate the process uses a digital surface model (DSM) to distinguish false positives and returns them as negative samples to the training [Kluckner et al., 2007]. However, not all potential false positives can be obtained in this way (e.g., road markings, rectangular transformer substations).

Another way of using HOG and LBP features is the Non-negative Matrix Factorization (NMF) [Mauthner et al., 2010]. The NMF shows an alternative to lower the dimensions

of the obtained feature vector and makes a SVM applicable. A feature vector with to many dimensions can pose problems to a SVM due to the curse of dimensionality.

The Harris interest operator is again utilized in a work by Gleason et al. [2011] aiming at very short processing times. It focuses on candidate regions which exceed a certain limit of the number of received Harris points. It is assumed that the background has a monochromatic color distribution and all regions that fit to that criteria are rejected. In the second stage features are calculated from eight sub-windows surrounding each candidate region. If one sub-window is accepted, the whole region is accepted. HOG features and Histogram of Gabor coefficients are applied together with a comparison of the following classification techniques: nearest neighbor, decision trees, random trees and support vector machines. The best results are achieved by the combination of Gabor derived histograms and random trees classifier. Moreover, the test data is of a very high resolution and shows only a small field of view from rural areas.

Finally, there is a system [Kembhavi et al., 2011] relying on three feature classes – HOG, the recently introduced color probability maps (CPM) and pairs of pixels (POP). The goal of the CPM is to represent the often homogeneously colored backgrounds of vehicles and typical vehicle colors in the center. The POP feature models the symmetric property of certain colored areas repeatable for many cars. All features concatenated, result in a feature vector of approximately 70,000 elements. At the end, the regression problem is solved by utilizing the Partial Least Squares (PLS) algorithm. False alarms are caused by rectangular car-like objects on top of buildings and road markings.

Region-based

A region-based technique such as the development of a sophisticated blob detector is carried out by [Lenhart & Hinz, 2006; Lenhart et al., 2008]. At first, vehicles with significant color features are detected by a color channel differencing method. From the remaining gray value images blob-like structures are extracted and the necessary threshold is dynamically determined depending on the road surface. The resulting elliptical blobs are evaluated in relation to their geometric moments and orientations of the surrounding ellipse. In addition, the ratio of major to minor axes of the ellipsis is used to avoid false positives.

Similar is the idea of another blob detector invented by Choi & Yang [2009]. They apply a mean shift segmentation in the Luv color space. Subsequently, the symmetry of the resulting blobs is examined by a filter based on complex valued Gabor functions. Additionally, the information of the shape is used. The shape of each blob is calculated by measuring the distance and orientation between the center of the blob and its surrounding edges. Often more than one blob is detected for the same car due to intensity differences from the front and rear windshields. The problem should be avoided by clustering blobs in a certain surroundings and with the same color values.

Likewise, there is an object-based classification technique starting with a multi-resolution segmentation based on region-growing [Holt et al., 2009]. Pixels are merged according to the following homogeneity parameters; scale, color-versus-shape, and compactness-versus-

smoothness. Thereby, the scale parameter controls the amount of heterogeneity of the segmented objects. The color-versus-shape parameter defines the extent to which overall homogeneity is defined by the spectral homogeneity. The smoothness-versus-compactness parameter controls whether segmentation results are optimized for objects with smooth borders or for those which have more compact shapes. All parameters have been implicitly determined by using training samples. Additionally, a spectral difference segmentation merges objects which are below a user-defined threshold of spectral similarity. This step enables modeling the road surface in order to distinguish between background and foreground. Besides, the RGB color values and its standard deviations, the remaining objects are classified using shape features like main direction, density and rectangular fit. In addition, texture features like density and mean of sub-objects are part of this technique. False negatives occur due to the inaccurate GIS database which is used to mask out city blocks and curbs. Cars close to the border of these areas are not detected.

Furthermore, the initial detection of shadow areas is the major aim of the approach of Wang [2011]. Firstly, a coarse-shadow map of the input aerial color image is generated by estimating a global threshold (Otsu method). Secondly, a connected component analysis is applied and the local threshold is calculated for every sub-region. In a third step, every pixel of a shadow candidate region is tested whether it belongs to the correct class or not. The assumption is that genuine shadow pixels have lower intensity values than their unreal neighbors, but both of their chromaticity values are similar. Additionally, it is assumed that the majority of genuine shadow pixels are connected. Afterwards a Harris corner response map and edge map of the RGB image are calculated at the locations of previously determined shadow regions. These interest points are further processed with the rotation invariant shape context feature descriptor. Finally, the resulting feature vectors are matched against reference feature vectors and it is accepted as a car if the matching cost is below a certain benchmark. A drawback of the approach refers to that the position of the cars is only roughly determined. Also cars in shady areas seem to be difficult to identify.

2.2.2 Multiple images

In this section all methods utilize information of more than one image. The arrangement is according to the branch of multiple images in Figure 2.1.

Difference images

The principle of difference images for a rough detection is appropriate to quickly get the overall traffic situation on highways. Two subsequent images are used by Reinartz et al. [2006] to calculate difference images. Two changes per moving car are returned. These changes have to be assigned to the first and the second image. Therefore, edges are extracted to distinguish whether the blob is due to a leaving or an arriving car. If the location of the contour coincides with the blob from the difference image, it is assumed that the object belongs to the current image. In the next step the obtained objects are refined by applying erosion and dilation. For high quality traffic analysis it is a prerequisite

to have a very accurate geocode and a very good co-registration. Also two subsequent images are used for a multivariate alteration detection (MAD) which results in a change image in which moving vehicles on roads are highlighted [Palubinskas et al., 2008]. The approach does not explicitly focus on the individual vehicle but on the traffic flow.

This strategy was taken up for twice the frame rate where the differences of three consecutive images are calculated [Xiao et al., 2010]. In parallel, a background learning and subtraction step is applied to detect slow moving or standing vehicles. Additionally, a co-registered road network delivers a vehicle behavior model and generates traffic pattern and additional regularization constraints. The graph matching algorithm combines the constraints with object-based vertex matching features and pairwise edge matching features into a single process. Finally, the overall association cost is minimized between current detections with the existing tracks.

Moreover, difference images of three subsequent images are used to extract the temporal change [Tuermer et al., 2011a,b]. Due to non-perfect co-registration many static regions have been extracted as well. A classifier based on HOG features and AdaBoost is used to examine the remaining objects.

Background model

In addition, the motion component is utilized in the approach of Reilly et al. [2010]. Firstly, the images are registered using Harris corner points and the SIFT descriptor, afterwards, outliers are removed by the RANSAC algorithm. Then a background model is calculated using simple median filtering for every 10 images. In a next step, the background is subtracted from the search image. Finally, remaining artifacts are removed by calculating the gradients of the background image and subtracted from the difference image as well. In general, all approaches placing reliance upon temporal change are quite accurate, but these methods only detect moving cars.

Stereo images

Based on the previous method, Pacher et al. [2008] add a calculated range image. Emphasis is on the determination of the ground area. Zebra crossings are utilized to get the height of the ground level. The same car extraction methodology is used to improve ortho-images and digital elevation models [Leberl et al., 2007, 2008].

2.3 Lessons learned and rationalization

Starting with the recapitulation of the ideas mentioned in the above presented car extraction approaches leads us to the following conclusions regarding the possible transfer to low frame-rate aerial imagery with a resolution of one or two decimeter.

Firstly, the employment of previous knowledge like the position of roads is a key factor to attain the best possible detection result. Information of road databases is often used

to limit the search space and restricts the extraction method only to areas belonging to roads [Holt et al., 2009; Kozempel & Reulke, 2009]. This has two major advantages: less calculation time and fewer false positives. Despite that fact, common road databases have a drawback concerning the accuracy of the positions of the roads and their borders. As the databases are mainly used for navigational applications, they are sufficiently accurate for the navigation task. But in the vehicle detection case we have to add a significant tolerance to the borders of the road to ensure that the whole road is examined. A better solution is to use road databases only for an ample extraction of the road. Additionally, the road segments can be extracted from the original image and not from the geo-referenced one in order to save calculation time.

A more reasonable step to deal with the dilemma of inaccurate road databases is the usage of DSMs (e.g., Kluckner [2011]). The ground level of densely populated city areas often belongs to roads or at least trafficable areas. Exceptions are bridges, flyovers, depressions or tunnel entrances/exits. However, these special areas can be determined by the utilized road databases or generally geographic information systems. Since cities are rapidly changing and the possession of global models is limited, it is suggested to calculate these DSMs directly before the vehicle detection procedure. Furthermore, to eliminate the calculation time which is necessary due to the geo-referencing step of the DSM generation process, disparity images are sufficient to distinguish ground from non-ground areas [Tuermer et al., 2012].

Normally, two overlapping subsequent images provide enough information for the disparity calculation. In the following chapter two different techniques are presented. One uses the position of the sensor which is obtained by GPS plus INS and the second one matches these two images using interest points only. Another advantage of disparity maps is that vehicle detection is not strictly limited to regions close to the center of the road, but also parking spaces which are slightly further away can be included. The presented strategy is initialized by a missing combination of methods in previous works. Many approaches have just been applied to single images as can be seen in Fig. 2.1. Thus, for this dissertation information from multiple images is utilized for disparity image calculation in order to exploit the 3-dimensional information for car detection. In addition, an automatic method is presented to separate ground from non-ground areas.

Moreover, the branch of single images in Figure 2.1 is further split in region-based and gradient-based methods. The region-based methods, on the one hand, often result in certain objects which than have to be classified by additional properties (e.g., geometry). The benefit is that usually the whole image is treated globally allowing existing interconnections between areas to be considered, such as green areas or driving surfaces. However, the utilized features are often rather simple. On the other hand, many of the latest gradient-based approaches, which are in the sub-branch of HOG features, rely on the sliding window technique – a technique which only operates locally (window size). Additionally, they use other complex high-level features but still the examined area is only local – the area of the window.

In this work, the combination of a region-based approach together with a high-level feature-based approach appears to be most straight forward and efficient. Both methods complement each other. In the case of the region-based step, a clustering of color

values is done. Subsequently, objects with certain geometries and shapes are selected. The high-level feature-based step is based on gray-value images, from which gradient magnitude and orientation are extracted in order to calculate HOG features. This feature is trained with an AdaBoost algorithm. In contrast to previous works, region-based and gradient-based features combined with disparity maps is suggested. Therefore, the novel region-based technique and a technique to automatically determine ground level are introduced.

In conclusion, this study offers the following major contributions:

- rapidly calculated disparity maps and the extraction of trafficable areas
- an effective region-based technique to select car candidates
- a combination of region-based and high-level features providing a high generalization in combination with low manual effort

The following research hypotheses are pursued. The combination of the region-based and the high-level feature-based methods is assumed to reduce the training effort. This may be possible because most of the non-relevant areas are excluded by the region-based method and the ground-area-determining method before the creation of the classifier. Generally, the parameter setting of the region-based technique should be less complex, and exhausting manual training steps like online training and back-porting of training samples should be avoidable.

3 Process Strategy

This chapter describes the methodological details of the suggested car extraction strategy. The order of the sections is according to their position in the processing chain. A short graphical overview is presented in Figure 3.1. It can be seen that the process starts with two subsequent overlapping images. Overlapping means they cover mainly the same area. Moreover, the single processing steps are indicated by rectangular forms. In addition to the label of each processing step the number of the corresponding section is included in the graphic as well.

3.1 Extraction of coarse road segments

The information of road databases or general GIS databases is frequently used to limit the search area in aerial images (e.g., Stilla & Michaelsen [2002]) or to control the search effort (e.g., Stilla [1995]). It has been shown that data from large vector maps (1:5000) or cadastral maps can be used in a very efficient way.

Example car detection approaches which try to extract areas belonging to roads are from Holt et al. [2009] and Kozempel & Reulke [2009]. However, common road databases distributed by commercial companies like Navteq [NAVTEQ, 1985], Tele Atlas [Tele Atlas, 1984] or nonprofit communities like OpenStreetMap [OpenStreetMap, 2004] have a drawback concerning their accuracy of the center-line and border positions of roads [Agamennoni et al., 2010]. On the one hand, road databases are mainly used for navigational applications for which they are sufficiently accurate. On the other hand, road databases are not suitable to determine the whole road accurately, for instance, without roofs from neighboring houses in urban areas or grass strips in rural areas. A slightly better performance can be sometimes achieved by road databases from governmental institutions like the Authoritative Topographic-Cartographic Information System (ATKIS) [AdV, 1996] provided by the Federal Agency for Cartography and Geodesy (BKG). However, some tests also showed a poorer reliability of ATKIS (deviations of up to 3.3 m) compared to NAVTEQ (average deviation 1.7 m, maximum 6.1 m) [Kozempel, 2012].

A second issue that comes up when talking about accuracy of road data bases is that original images after direct georeferencing (e.g., ortho-image) also do not have a highly accurate geocode. This is due to limited accuracy of GPS/IMU inside the plane and calibration errors. When summing up both errors (geocode of ortho-image and database) the desired center-line of the road can be several meters away from its real position. However, this argument is only valid for the real-time case because the accuracy of the geocode can be enhanced if enough time for a post-processing is available.

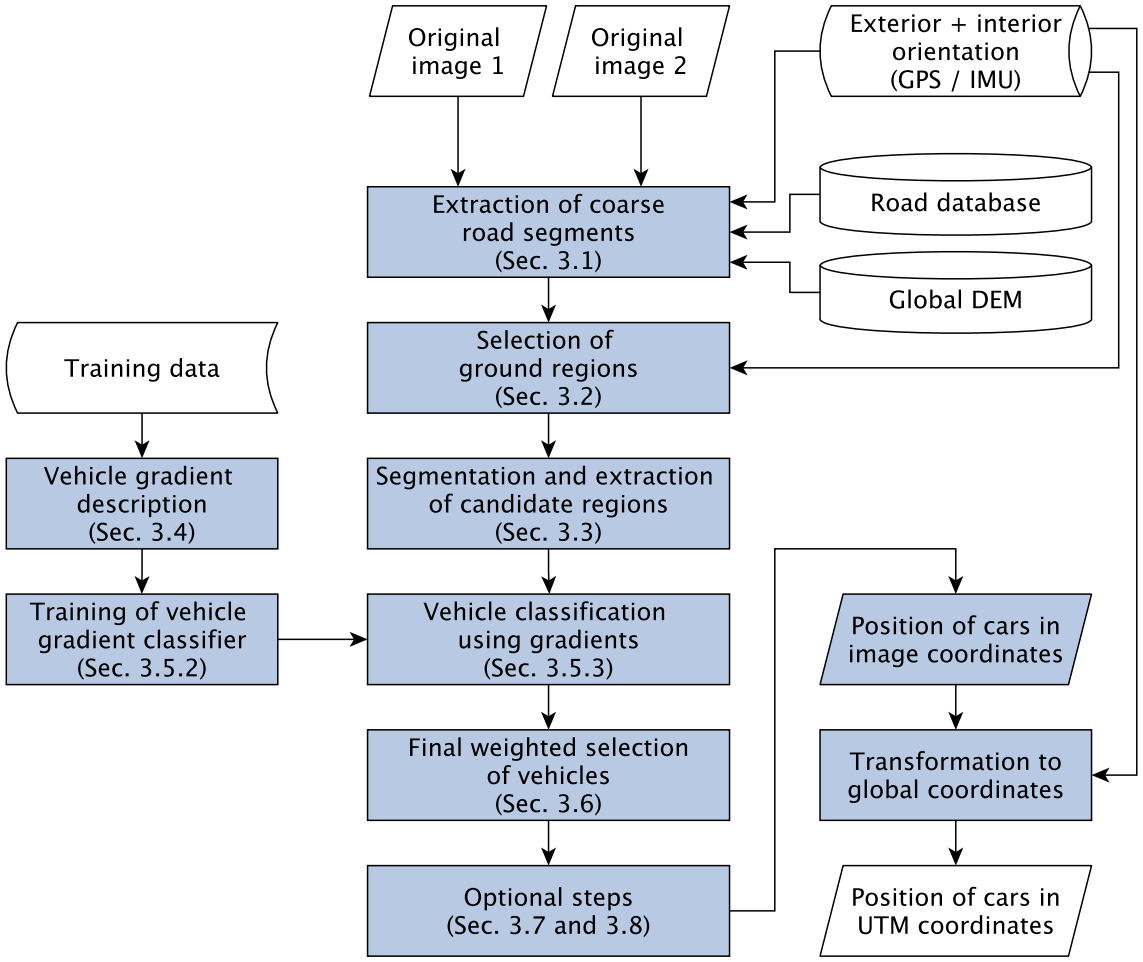


Figure 3.1: Workflow of presented car extraction strategy. The databases of the roads and the global DEM are available in advance. Moreover, the training dataset is also available before the images are received.

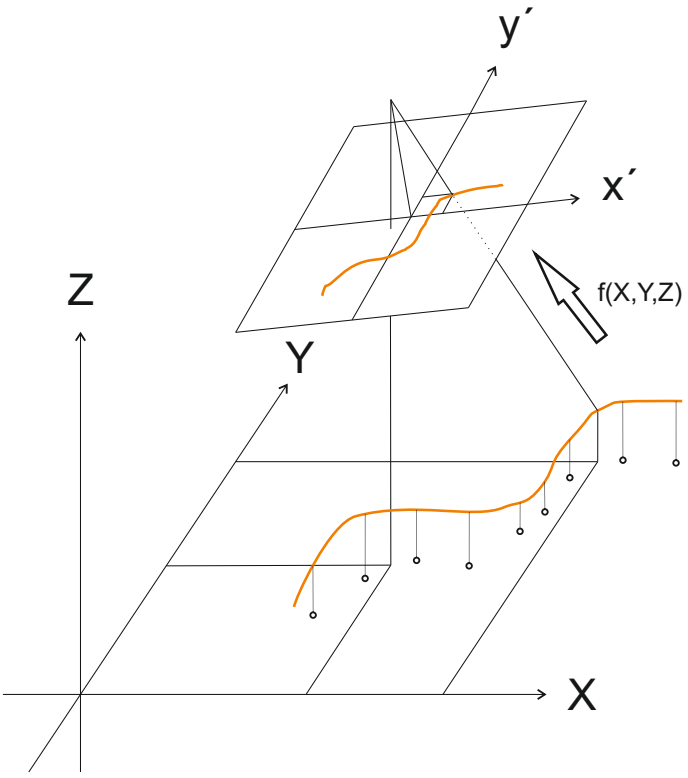


Figure 3.2: Projection of road segment from road database to the original image. The coordinate system X,Y,Z is from the road database and the DEM, while the coordinate system x', y' is from the image. The function f(X,Y,Z) is described in Equations 3.1 and 3.2.

Conclusively, using current road databases is not a sufficient solution to accurately limit the search area. In addition, the limitation using road databases could also have drawbacks. Considering the fact that when only roads are extracted, cars in parking spaces in the surrounding area, cannot be detected either.

Hence, the idea pursued in this dissertation is to extract road segments plus a generous buffer zone. Also in the case of extreme inaccuracies all roads and the vehicles on them should be preserved in the remaining areas. In contrast of using the whole image, the benefit is still the reduction of the calculation time for further processing steps and the risk reduction of false positives in areas with car-like objects. However, it is not necessary to use ortho-images with geocode. This step would lead to further time consumption, and depending on the resampling algorithm, also to a worse image quality. The proposition is to project the road segments in the original image as shown in Figure 3.2.

A position in the image (x', y') can be calculated with the collinearity equation:

$$x' = x'_0 - c\left[\frac{r_{11}(X - X_0) + r_{12}(Y - Y_0) + r_{13}(Z - Z_0)}{r_{31}(X - X_0) + r_{32}(Y - Y_0) + r_{33}(Z - Z_0)}\right] \tag{3.1}$$

$$y' = y'_0 - c\left[\frac{r_{21}(X - X_0) + r_{22}(Y - Y_0) + r_{23}(Z - Z_0)}{r_{31}(X - X_0) + r_{32}(Y - Y_0) + r_{33}(Z - Z_0)}\right] \tag{3.2}$$

where the interior orientation consists of the coordinates of the principal point x'_0, y'_0 and the calibrated focal length c. The exterior orientation is included by the coordinates of the projection center X_0, Y_0, Z_0 (obtained from GPS) and the rotation of the camera r_{ij} (obtained from IMU and boresight misalignment [Kurz et al., 2007; Lee & Yilmaz, 2011; Kurz et al., 2012]). X, Y are the coordinates which are received from the road database while Z is obtained from a global DEM.

Due to the fact that a high accuracy is not necessary at this step (the enhanced extraction of trafficable areas is explained in Section 3.2), a coarse DEM for example of the SRTM mission can be used [USGS, 2000]. The absolute height error (90% error) of the SRTM measured in Europe and Asia is 6.2 m and in North America 9.0 m [Rodriguez et al., 2006]. Other sources report an error of 4.07±0.47 m in Catskill Mountains (New York, USA), which is significantly better than indicated in the specification (16 m) [Gorokhovich & Voustianiouk, 2006]. These previously mentioned values refer to the DEM obtained from C-band interferometric radar data but there is also a DEM based on the X-band. These two DEMs can be combined to further enhance the accuracy. The standard deviation of the differences of the combined DEM and an absolute elevation reference in southern Germany is 3.4 m [Hoffmann & Walter, 2006].

3.2 Selection of ground regions

The intention is to calculate a disparity image and to exclude areas above ground level where cars are found very unlikely. Bridges, flyovers, tunnel entrances and exits are special cases and have to be treated differently. To speed up the calculation we use the coarse road segments which we cut out using the road database and calculate the disparity image only for these two consecutive segments. Finally, the ground area of the disparity images is automatically determined.

3.2.1 Calculation of disparity image

In the following paragraphs a method for calculating the disparity image is presented. This is split up into obtaining the orientation of the two cameras and the calculation of the epipolar images, and matching of corresponding pixels from image 1 to image 2 with the semi-global matching algorithm.

Orientation of the stereo images

The procedure of calculating the orientation of the images starts with calculating interest points in both images. Popular ones are, for instance, Harris [Harris & Stephens, 1988] or Foerstner points [Förstner & Gülch, 1987]. The latter are utilized here due to their better performance concerning distinctness, invariance, stability, uniqueness, and interpretability [Rodehorst & Koschan, 2006]. Although, the evaluation of interest point operators depends on the scene and the implementation. Other possible interest points

which have advantages – for example rotation invariance – are the SIFT [Lowe, 2004], the SURF [Bay et al., 2008] or the BRISK [Leutenegger et al., 2011] operator. Also a combination of SIFT and Foerstner points is possible and has been alternatively evaluated. This results in a technique which combines the robustness of the SIFT and the location accuracy of the Foerstner operator [Heinrichs, 2011].

The geometry of the stereo setup is figured out by a matching of the previously generated interest points. More precisely explained, gray values of a certain area around the interest points are matched using normalized cross-correlation. The optimal setting of matching points is obtained by filtering with the RANSAC algorithm [Fischler & Bolles, 1981]. Goal is to iteratively find the setting where a maximum of interest points is conform with the epipolar constraint (minimum distance of corresponding points from the epipolar line). In addition, lens distortions are considered by using a non-linear camera model with parameters of the interior orientation.

Two different ways are shown to finally obtain the orientation of the cameras – relative or exterior orientation. The exterior orientation is the combination of relative and absolute orientation. The first way does not utilize additional information, while the second exploits the navigation data of the aircraft. Navigation data are the position obtained from the GPS and the rotation of the IMU sensor.

Relative orientation without navigation data After the previous steps a set of corresponding points is available. These points are used to estimate the fundamental matrix F with a non-linear iterative algorithm based on the Maximum Likelihood Estimation. The algorithm is described in Hartley & Zisserman [2010] (Algorithm 11.3, The Gold Standard algorithm for estimating F from image correspondences). The matrix F consists of a matrix of translation and a projective transformation corresponding to the corrections of the first camera. After determining the F matrix, the two stereo images are resampled considering the epipolar constraint. After the transformation, corresponding epipolar lines are co-linear. The resampling is done with a bi-linear interpolation algorithm. The epipolar images allow us to search for the match of a point in image 1 along the corresponding epipolar line in image 2 [Kraus, 2007]. The benefit of the epipolar geometry is that it reduces the scope to a one-dimensional correlation problem.

Exterior orientation with navigation data The second way is used when navigation data are available. In order to utilize the additional information, a bundle adjustment is applied [Triggs et al., 2000]. This procedure is assumed to be more accurate because then the position and the rotation of the cameras from GPS and IMU can be introduced to the bundle adjustment as additional observations. The bundle adjustment is done to estimate the exterior orientation, which is then used to calculate the epipolar images.

Semi-global matching

The disparity images are then calculated based on the epipolar images utilizing the semi-global matching (SGM) algorithm [Hirschmueller, 2008]. The basic steps of the stereo vision method have the following properties [d'Angelo & Reinartz, 2011]:

Matching cost computation The Census transform [Zabih & Woodfill, 1994] is used to compute the similarity value of two matched pixels. It is based on small windows and is considered very robust in the case of discontinuities [Hirschmueller & Scharstein, 2009]. For a further computation of the matching costs the Hamming distance [Hamming, 1950] is used.

Aggregation of cost and disparity computation Due to the global algorithm an energy function is optimized. The energy $E(D)$ is defined as [Hirschmueller, 2008]:

$$\begin{aligned} E(D) = \sum_{p} (C(p, D_p) + \sum_{q \in N_p} P_1 \cdot T \cdot [|D_p - D_q| = 1] \\ + \sum_{q \in N_p} P_2 \cdot T \cdot [|D_p - D_q| > 1]) \end{aligned} \tag{3.3}$$

where D is the disparity map, the pixel matching costs for each pixel at location p in the first image and its corresponding pixel in the second image (given by the disparity image D_p) is defined by function C. The next two terms add penalties (P_1, P_2) in the case of small (e.g., 1 pixel) or larger disparity changes in neighborhood N_p. To this end, T is set to 1, if the argument is true or to 0, if not.

Refinement of disparities Sub-pixel accuracy can be obtained by fitting a local parabola to the aggregated costs close to the minimum. Additionally, to remove outliers, pixels of image 1 are matched to pixels of image 2 and vice versa. The disparity is rejected if there is no consistency.

3.2.2 Determination of ground areas from disparity image

The ground area of the disparity image is automatically determined using a technique from the field of minimum error thresholding [Kittler & Illingworth, 1986]. The intention is to iteratively find the best separation between two classes (ground and non-ground). The method was developed under the assumption that the part of the image which is cut out, with the pre-knowledge coming from road databases, has two main classes in densely populated urban areas. These classes are roofs of high buildings and roads/pedestrian paths.

The algorithm works globally on the selected road segment and can be mathematically expressed as follows.

$$\begin{aligned} T_{opt} = \arg\min\{1 + 2[R_1(T)\log\sigma_1(T) + R_2(T)\log\sigma_2(T)] \\ - 2[R_1(T)\log R_1(T) + R_2(T)\log R_2(T)]\} \end{aligned} \tag{3.4}$$

where T is the examined threshold and $\sigma_1(T)$, $\sigma_2(T)$ are foreground and background standard deviations. The parameter R_i is calculated with Equation 3.5.

$$R_i(T) = \sum_{g=a}^{b} h(g) \tag{3.5}$$

with $a = \{0|i = 1\} \vee \{T|i = 2\}$, $b = \{T-1|i = 1\} \vee \{n|i = 2\}$, n is the number of intensity values, and $h(g)$ is a histogram of the elevation values. The algorithm walks through every possible threshold and evaluates it with the criterion function T_{opt}. A comparison of this method to others can be found in Sezgin & Sankur [2004].

3.3 Segmentation and extraction of candidate regions

In aerial images of one or two decimeter resolution cars mostly appear, simply described, as rectangular-like objects having a similar shape with a certain tolerance depending on the genre of the specific car. Exceptions are, for instance, partly occluded vehicles by trees or other objects with overhang. Also the perspective projection (central projection) of aerial images leads sometimes to occlusions. Objects higher than cars, like buildings, 'fold back' and obstruct the view.

The approximate average size of a car is 4.5 m length [Kienzle, 2001] and 2 m width in the real world. Based on that knowledge the car size in the image can be easily calculated corresponding to the image's GSD. After a successful segmentation it is possible to examine the segments obtained according to their size and shape. The segmentation result will be more sophisticated when color information (RGB channels) is used instead of gray values only.

Usually, two-tone or multi-tone colored vehicles can be observed quite rarely. The very vast majority of all cars are painted in a single color. And within the group of single-tone colored vehicles more than two-thirds have no color as such; these cars are black, white or gray. The tendency becomes clear when looking at the colors of newly registered vehicles in the year 2012 [DuPont, 2012] (see Figure 6.1). Statistics of vehicle colors in the case of Germany only are provided by Kraftfahrt-Bundesamt [2011, 2012].

Segmenting single-tone colored objects should be easy. However, in practice the problem is that although a car is painted in a single color it appears in the image as an object having slightly different tone variations due to varying illumination. This fact makes the segmentation process more challenging. Additionally, objects like the windshields or the lights are often in another color anyway. A solution that eases segmentation is an additional smoothing step which leads to more homogeneous colored objects. Also the

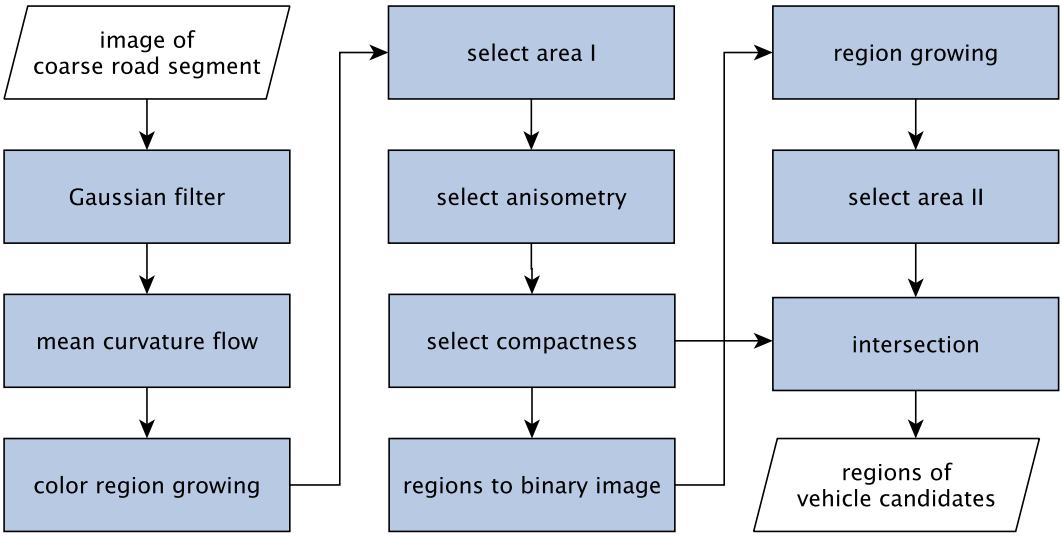


Figure 3.3: Workflow of car candidate selection.

influence of general noise is reduced. The smoothing can be carried out more effectively when the pixel size of the desired objects is known. Because as consequence the choice of an optimal sigma for Equation 3.6 or Equation 3.7 is easier.

The segmentation step is generally a grouping of pixels with similar RGB intensities. In order to get objects based on the same color, a region growing algorithm is applied. In a further step these obtained regions are filtered according to their properties. The goal is to identify certain objects by their typical shapes and forms; also involving the number of pixels.

Smoothing of images has the major consequence that it removes high frequencies. These high frequencies are responsible for the visibility of small details. Small details in aerial images with a resolution in the lower decimeter range are, for instance, street lamps, traffic lights or pedestrians. But also parts of larger objects, like in the case of a car, head lights or sliding roofs belong to this group. After removing these high frequencies the segmentation returns fewer objects which can be better classified. For instance, objects with a significantly different aspect ratio such as a car or car queues can be rejected.

The proposed method is also very powerful when it comes to rejection of road markings. These objects often pose difficulties to other algorithms because they show strong gradients at their borders. Additionally, the distance of parallel located road markings is often only slightly wider than the width of a typical car.

An overview of all the single sequential steps is shown in Figure 3.3.

3.3.1 Smoothing and mean curvature flow

Gaussian filter

At the beginning of the segmentation step it is necessary to smooth the image and remove very high frequencies. Therefore, firstly a Gaussian blurring is applied to reduce eventual present noise which could have a disturbing effect in the subsequent process. The discrete Gauss function G is defined as:

$$G_\sigma(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{3.6}$$

where x, y are the two image dimensions and σ is the standard deviation. In practice, σ is chosen according to object size in the image. With respect to the GSD of the image, the filter size is chosen.

Mean curvature flow

The second smoothing step is more crucial and has greater impact. Generally, cars appear differently in aerial imagery. Some show fine structures and others appear as dark blobs. The smoothing is an attempt to generalize them in such a way so that the regions can be better classified after the region growing procedure. In the best case, the result is that cars remain just as a single-colored patch. Additionally, often road markings can be removed from the further search process (see Chapter 5.1.2).

Three parameters have to be set for the application of this function [MVTec, 2012]. The parameter σ (sigma) of the Gaussian convolution kernel (see isotropic smoothing in Section 3.3.1); also necessary are time step θ and number of iterations i. The parameter σ is used for an Gaussian smoothing as done in the previous step and is not necessary anymore. The two other parameters are explained in the following lines which explain the problem of mean curvature flow.

Generally, our image is expressed by function $u(t, x, y)$ at time $t_{start} = 0$ which is $u(0, x, y)$. The aim is now to determine the temporal change due to the curvature flow which is given by Equation 3.7 [Chen et al., 1991].

$$\frac{du}{dt} = \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right)|\nabla u| \tag{3.7}$$

It can be seen as an initial value problem. Here ∇u is the spatial gradient of u, and $\nabla u/|\nabla u|$ is the unit normal to the level surface of u. Hence, $\operatorname{div}(\nabla u/|\nabla u|)$ is its mean curvature.

Finally, the numerical technique named finite element method (FEM) is used in order to calculate $u(t, x, y)$ at a certain time $t = t_{end}$. That means all partial derivatives are approximated by a difference quotient. The central difference quotient is used for the derivatives of x and y. The forward quotient is used for the derivation of time t because $u(-\theta, x, y)$ is not known.

The derivative of the spatial coordinates can be evaluated at certain locations only due to the fact that an image is a discrete function. These certain locations are given by an increment in pixels. The same principle is valid for the derivative of time which is also calculated at discrete time steps only. This increment is given by the parameter θ. It is only calculated at a finite number of time steps which are given by the parameter i.

The solution $u(t = t_{end}, x, y) = u(i \cdot \theta, x, y)$ is iteratively calculated: Using the given initial value u(0,x,y) enables us to calculate $u(\theta, x, y)$. Moreover, the determination of $u(2 \cdot \theta, x, y)$ can be done with $u(\theta, x, y)$ until $u(i \cdot \theta, x, y)$ is reached.

The partial derivative of u to t at the location $i \cdot \theta$ is expressed by Equation 3.8.

$$u_t = \frac{u((i+1) \cdot \theta, x, y) - u(i \cdot \theta, x, y)}{\theta} \tag{3.8}$$

This results in Equation 3.9 according to Aubert & Kornprobst [2006]:

$$u((i+1) \cdot \theta, x, y) = \frac{u(i \cdot \theta, x, y) + \theta \cdot \sqrt{u_x^2 + u_y^2} \cdot (u_{xx} u_y^2 + u_{yy} u_x^2 - 2 u_x u_y u_{xy})}{(u_x^2 + u_y^2)^{3/2}} \tag{3.9}$$

where u_x, u_y, u_{xy} are the derivatives which are approximated by the difference quotient like in Equation 3.10.

$$u_x = \frac{u(i \cdot \theta, x + \Delta x, y) - u(i \cdot \theta, x - \Delta x, y)}{2 \Delta x} \tag{3.10}$$

with Δx being the increment in x direction. The other directions are approximated in the same way.

Benefit of this method is a smoothing along and not perpendicular to the edges of the image. It can be also described as contours that move in the direction of the gradient. A further description of this topic can be found in Crandall & Lions [1996] and [Clarenz et al., 2003].

3.3.2 Region growing and selection of vehicle candidate regions

The following algorithms are applied to the processed image from the previous Section 3.3.1.

Color region growing

Firstly, an unsupervised clustering is done and the image is segmented in several regions. If two neighboring pixels have a distance lower than a certain threshold they will be agglomerated. The following schema for calculating the distance is used:

$$U = \sqrt{\frac{1}{3}\sum \left[\begin{bmatrix} R_a \\ G_a \\ B_a \end{bmatrix} - \begin{bmatrix} R_b \\ G_b \\ B_b \end{bmatrix} \right]^2} \tag{3.11}$$

where U is the Euclidean distance and R, G, B are the red, green, blue channels of neighbouring pixels a and b.

Select area I

The obtained regions are filtered with regard to their geometric properties. Regions consisting of fewer pixels than a certain limit are selected. The limit is set to the maximum size of a car aimed to detected – plus a tolerance. At this point there is no minimum limit to reject regions which are only sparsely populated. Because, one and the same color often looks different due to different illuminations, and thus this effect might lead to more than one region per car.

Select anisometry

A second filter step selects regions concerning their anisometric properties. The anisometry An is derived from the division of the major axis Ra and the minor axis Rb of the region (Equation 3.12).

$$An = \frac{Ra}{Rb} \tag{3.12}$$

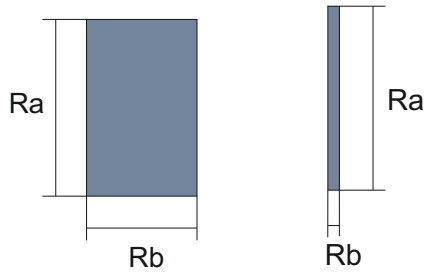
The ratio delivered by the anisometry measurement is useful to distinguish between regions having the same number of pixels but a different shape. Regions that are too thin or too long, and would not match to a vehicle can be rejected. In practice these regions are often triggered by boarders of roofs or road markings. The principle schema can be seen in Figure 3.4.

Select compactness

As last filter step for the preliminary car candidates, the compactness of the regions is examined. The compactness C of a region is calculated by the following mathematical expression (Equation 3.13).

$$C = \frac{L^2}{4A\pi} \tag{3.13}$$

where L is the length of the contour of the region. The distance between pixels parallel to the coordinate axis is counted as 1 while the diagonal distance contributes with $\sqrt{2}$ to the overall sum. The parameter A is the area of the region which is simply the sum of all pixels.



(a) (b)

Figure 3.4: Visual description of the anisometry measurement. Region (a) would be accepted due to its fitting ratio of Ra and Rb, while the region (b) does not match that criteria and is rejected.

A benefit of the compactness measurement is that there is a sensitivity to roughness and gaps incorporated. This is also useful to reject regions which are not typical for a car.

Regions to binary image

The remaining regions are now transformed to a binary image but this is not the final mask yet. The step is necessary to apply the second region growing.

Binary region growing

The second region growing step is only applied in order to form one single region from several directly neighboring regions. Single regions without neighbors still remain single regions and are not agglomerated. The applied method is just a binary region growing algorithm and the formula is equal to Equation 3.11 but only for one channel.

Select area II

After the second region growing, the agglomerated regions are examined according to their size once again. Regions below a certain number of pixels are rejected. This procedure allows us to get rid of very small artifacts which come originally from vegetation or from roof ridges.

Intersection

Unfortunately, due to the second region growing and the "Selected area II" step regions are generated that have already been excluded previously. Therefore, an intersection of the current status with the previous status (after "Select compactness") is applied. This assures that the result is not worse than it was at an earlier step in the processing chain.

3.4 Description of vehicles by gradients

The description of a car can be done using gradients. Many cars have more or less typical edges which lead to gradients and can be used for a general description. The following section shows which edges are useful for classification of cars and how they are processed to get a high level feature like HOG. There are different categories of vehicles, which have different edges with slightly different curvatures in the image.

3.4.1 Calculation of gradients

Edges that can be expected from a car in aerial imagery are shown in Figure 3.5. In reality, often not all edges are present at once. However, the key attribute of a car seen from above is the surrounding contour with its rectangle-like shape and rounded corners. If this contour is not present it is very likely not to be a car. The only exception could be a partly occluded car (e.g., by trees).

Other edges appear at the windshield and the rear window. Whereas the windshield is often larger and more frequently present in the view from above, many car types like vans or station wagons have a very steep rear window which can be hardly seen from above. Another problem is the color of cars. While bright cars offer a good contrast between the color and the typical dark windows, dark cars do not. In that case the windshield does not provide any useful edges. Further edges can be triggered by road markings which may occur at random positions around the car. Although there are standards for road markings, it is very difficult to find a good general model. Some road markings are interrupted, others are continuous and even some areas are transversely lined to indicate a certain traffic rule. The magnitude of edges from road markings is usually very high due to the good contrast which is given by the white color of road markings in contrast to the gray road surface. Also the impact of shadow has to be considered. Shadow can be problematic due to edges which arise by reason of boundary lines of the shadow (especially of the umbra). Firstly, shadow only appears under illumination; secondly, the shadow size depends on the position of the sun (time of day, time of the year, orientation). Moreover, the location is related to the orientation of the car to the sun. Furthermore, if a car is already in the shadow of a house for instance, its own shadow is no longer visible. Finally, edges that are often neglected refer to the front lights. Especially, modern cars have large headlights which can be sometimes seen from above. If these edges exist they have usually a high magnitude due to the good contrast of the inherently white headlights.

Gradient magnitudes and gradient orientations in gray level images can be obtained by various operators. Popular ones are the Prewitt [Prewitt, 1970], the Sobel [Sobel, 1970] and the Scharr [Scharr, 2000] operator where the two last two additionally include amplifying factors. Problems in case of these filters could be the anisotropy; in other words a stronger response from horizontal and vertical edges than from diagonal ones results. The Scharr operator shows a significantly improved rotational invariance compared to the Sobel operator [Kroon, 2009].

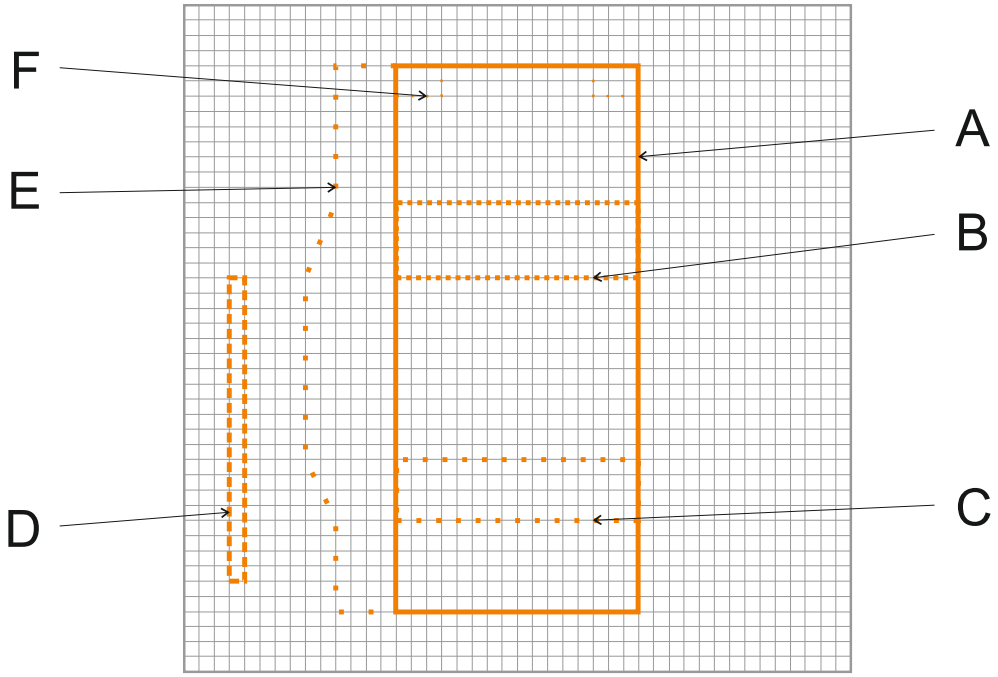


Figure 3.5: Typical edges that can be expected from a car in aerial imagery. (A) The surrounding contour is the key characteristic which can be observed for every car. Exceptions are only partly occluded ones. (B) The front windshield is also often visible except for some dark cars. (C) Gradients of the rear window occur more rarely due to different car types. Some car types have almost perpendicular rear windows and cannot be seen from above. (D) Road markings can occur at every location around the car. (E) On a sunny day also a boundary line of the shadow has to be expected. (F) Some cars provide edges due to their usually bright headlights.

Another possible solution is the introduction of an additional filter to detect diagonal edges known as a compass filter. This is done with the implementation of the following operators. The Robinson [Robinson, 1977], the Kirsch [Kirsch, 1971] or the Frei-Chen [Frei & Chen, 1977] operator are isotropic.

Furthermore, some operators are more sensitive to noise and others have a better generalization ability which is also related to the filter size. Typical sizes are 3 by 3 or 5 by 5. Sometimes it is also reasonable to apply a smoothing filter (e.g., Gauss filter) in advance.

However, the suggestion of Dalal & Triggs [2005] in the case of face detection is that a simple filter like the Prewitt performs best. This statement is also valid concerning vehicle description in aerial images, as own tests showed, and thus can be confirmed. The other operators cannot really compete except for the Sobel operator. The Sobel operator's attribute is putting emphasis on stronger edges which leads to a better generalization ability. However, it is in general not recommended because the weighting leads to the loss of the smaller gradients. The previously mentioned generalization ability leads sometimes to fewer false negatives but it can also lead to more false positives. If a car is extremely generalized only the surrounding contour will remain, which is more or less similar to a simple rectangle.

The gradients in x and y direction are obtained by two convolutions of the gray value image. Finally, the best performing operator is the Prewitt filter which can be mathematically expressed as shown in Equations 3.14 and 3.15.

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{bmatrix} * I \tag{3.14}$$

$$G_y = \begin{bmatrix} +1 & +1 & +1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} * I \tag{3.15}$$

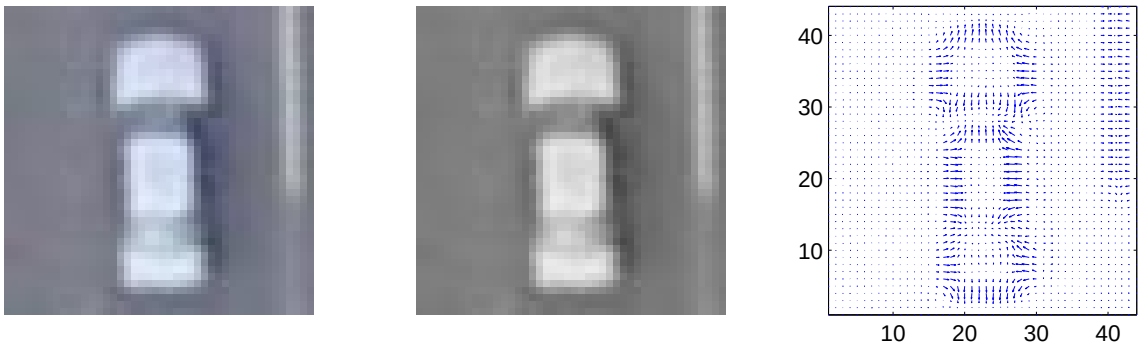
where G_x are the high spatial frequencies in horizontal direction and G_y in vertical direction. The gray value image is I. The gradient magnitude is given by Equation 3.16 and the angle of orientation of the edge is given by Equation 3.17.

$$|G| = \sqrt{G_x^2 + G_y^2} \tag{3.16}$$

$$\theta = \arctan \frac{G_y}{G_x} \tag{3.17}$$

The border treatment, which is necessary for the gradient calculation, has not such a great impact. It can be done by setting all border values to zero, replicate or mirror them. The final decision was to replicate the boarders.

An example of resulting gradients from cars in aerial imagery calculated by the Prewitt operator is shown in Figure 3.6.



(a) (b) (c)

Figure 3.6: Example of Sobel operator application. (a) Original RGB image showing an average car. (b) Same car after RGB to gray value conversation. (c) An example of the application of Equation 3.14 and Equation 3.15. The blue arrows show the orientation and magnitude of the gradients.

Also the Prewitt filter response for an intermittently rotated object is show in Figure 3.7. It can be seen that the slightly remaining anisotropy of the filter, which is inherent for gradient filters, has a minor impact only.

3.4.2 Calculation of histogram features

The HOG feature was inspired by the previous work of Lowe [1999] who invented the SIFT key point method. Part of the SIFT method is the SIFT descriptor. The idea to use the SIFT descriptor as descriptive feature for object recognition was introduced by Dalal & Triggs [2005]. The main principle of this feature is binning the magnitude of the gradients to a histogram according to their orientation. Thus, the exact position of the gradient gets lost but the position of the area from which the feature is calculated remains. The histogram provides a certain generalization and the number of elements in the feature vector is reduced. A detailed description of the HOG feature can be found in Dalal [2006].

Nowadays, there have been a number of descriptor variants presented where the roots are still in the SIFT descriptor. One of the latest further developments is the CHOG descriptor where the C stands for the word compressed [Chandrasekhar et al., 2009, 2012]. It uses a Huffman coding to compress the information of the gradients and lowers the feature vector dimension. It is recommended to use this feature when the data of the feature has to be transmitted via low bandwidth. For an evaluation of other popular local features up to the year 2005 please refer to Mikolajczyk & Schmid [2005].

The original HOG feature calculation starts with a grid of overlapping blocks. Each block contains a grid of cells. The weighted votes for gradient orientation are accumulated in each cell. The blocks are used to normalize the contrast. According to the block form there are R-HOG and C-HOG features which stands for rectangular and circular, respectively.

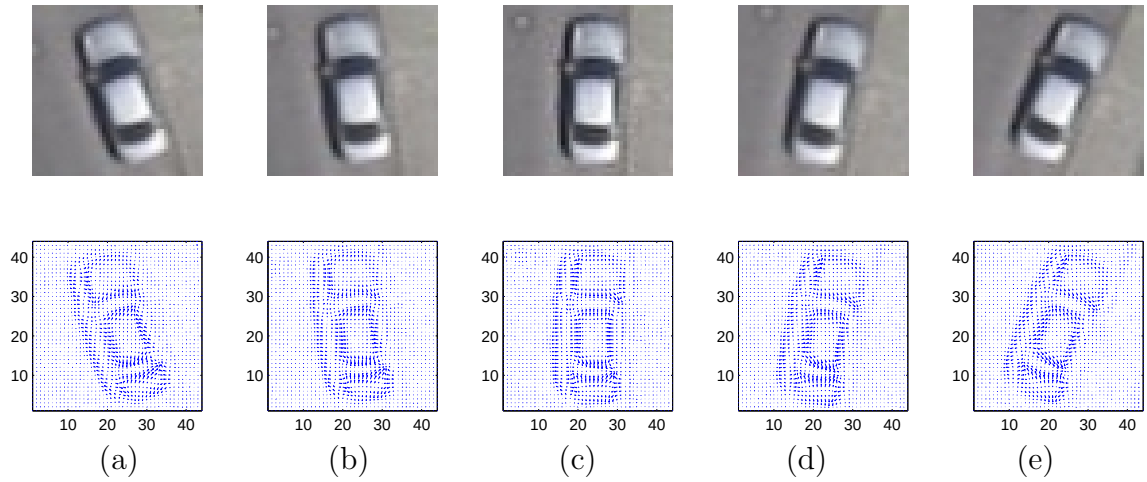


Figure 3.7: Same car in different orientations. The gradient filter, which was used for demonstration purposes, is the Prewitt filter. It can be seen how the gradients change due to the rotation and that the slightly remaining anisotropy of the filter has a minor impact only. The size of the image is 44 by 44 pixels and thus the unit of the coordinate axes is pixel.

In our approach we also utilize the SIFT descriptor with rectangular cells. But in contrast to the original HOG feature we have overlapping cells of different sizes. These cells are shifted over the whole patch as described in Figure 3.8. The usage of different cell sizes leads to more features and can increase the detection quality. The optimal cell sizes depend on the GSD of the desired object. In general, it is not necessary to be restricted to only a few sizes because the detection quality won't be affected by too many features; especially in the combination with a machine learning algorithm like Boosting (Section 3.5). The only limitation of the increasing number of features is that these have to be computationally handled. That means the estimated training time will increase and the hardware configuration (e.g., memory) should be appropriate. The runtime of the final classifier is not affected.

The following cell sizes in pixels are used $\{4\times4, 6\times6, 8\times8, 10\times10, 16\times16\}$. Finally, when the features of all the cells are calculated each training sample provides 6280 features for a training sample sized 44×44 pixels (44×44 is a suitable size for a car in aerial imagery with 13 cm GSD). Smaller cells are not reasonable because 2×2 pixels do not have enough significant information. The upper limit simply depends on the object's resolution. The impact of the different cell sizes is shown in Section 4.3.1. Furthermore, the utilized sample classifier is shown which consists mainly of 4×4 sized features (Figure 4.7).

Each histogram is normalized. Generally, a global contrast normalization can improve the performance as well [Dalal & Triggs, 2005]. However, the algorithm applied here is without global contrast normalization. A possible concept is to detect shadow areas and treat them differently [Makarau et al., 2011].

Another important parameter is the decision of how many bins each feature consists. The experiment for human detection showed that an optimal number of bins is 9 [Dalal & Triggs, 2005]. Own experiments showed that there is no difference between 9 and 8 bins. Even the difference between 4 and 8 bins is not very high. However, in the further

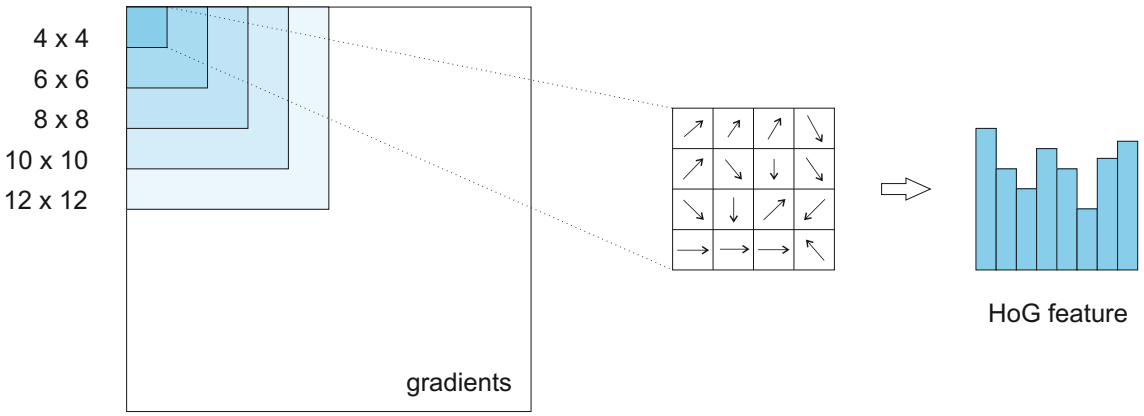


Figure 3.8: Schematically explanation of the utilized histogram feature. The process starts with the calculation of gradient magnitude and gradient orientation based on a patch from a gray value image. Finally, these gradients are transformed to the histogram. The selection of the correct bin is according to the orientation. The related magnitude of the gradient is added to the height value of the bin. The final feature is the resulting histogram.

approach 8 bins are utilized. In a very time critical environment the recommendation is to go for 4 bins only. The information stored in 4 bins can be enough to describe the typical rectangular shape of a car which is the mainly used attribute.

Furthermore, another important issue is if the histograms are calculated for 180 degrees only or for the total 360 degrees of the orientation of the cars. In the first case training cars are not oriented in one direction only but the diametrically oriented cars are also used. In contrast, the second case uses only cars strictly oriented in the same direction. As I stated before, the training data should be as homogeneous as possible, hence the optimal solution would be to use 360 degrees and thus only cars which are oriented in one single direction.

However, when applying a detector based on single oriented cars to an image where cars are included in two diametrically orientations, all cars of the other orientation are detected as well. The reason is the anyway already existing strong generality of the detector due to the fact that cars can never be completely homogeneous. Consequently, the detector trained for one orientation is also applied for the opposite orientation.

3.4.3 Car model and similarity measurement

The calculated histograms need to be compared. Therefore a schema has to be applied which returns a distance and enables us to evaluate the distance between two histograms. A common way do do that is using the Bhattacharyya distance D_B [Bhattacharyya, 1943].

$$D_B(P,Q) = -ln(BC(P,Q)) \tag{3.18}$$

$$BC(P,Q) = \sum_{i=1}^{n} \sqrt{\sum p_i \sum q_i} \tag{3.19}$$

where P, Q are two histograms, p_i, q_i are the corresponding bins (elements) of the histograms, and thus $\sum p_i$ or $\sum q_i$ is the magnitude of the respective bin of the histogram.

The D_B is positive and symmetric however it violates the triangle inequality [Kailath, 1967]. A fact which would not interfere but the Hellinger distance D_H instead meets all axioms that are necessary for the definition of a metric [Comaniciu et al., 2003] and is used in the following.

$$D_H(p,q) = \sqrt{1 - BC} \tag{3.20}$$

3.5 Vehicle gradient classifier

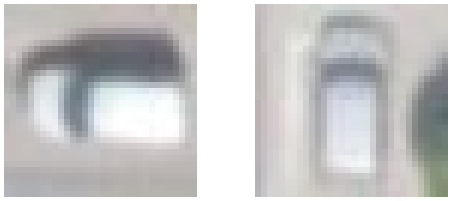
The training procedure of the gradient-based classifier is assigned to implicit methods [Gomes et al., 2009]. Within that group a further division is possible between unsupervised and supervised classification. The first mentioned is the generic term for methods using statistics of the data itself to separate the feature space. In contrast, the second mentioned which uses extra training data to figure out the setting of the parameters. This thesis deals in the following section with a supervised classification and the Boosting algorithm. It is going to be explained how the selection of training data and its processing is carried out. Subsequently, relevant AdaBoost variants are described and compared.

3.5.1 Selection of training data

The success of the final classification result is highly ascribable to the preparation of the training data. It is insufficient to cut out the vehicles haphazardly and forward that data to the Boosting algorithm. The explanation is as follows. All training data, positive ones as well as negative ones, provide a set of features (which are calculated as described in Section 3.4.2). These features are the origin for determining the dividing lines within the feature space. These dividing lines are the base of the parameters of the subsequent classification. If the positive features are very heterogeneous it is more difficult to separate them from the negative class of features. At a certain point it will not be possible to divide the feature space accurately and you have to make the decision whether to accept more false positives or more false negatives.

The general heterogeneity of the vehicles in aerial images is connected to the following reasons:

- Vehicles have *different dimensions* which depend on their genre
- Vehicles are *differently oriented*, even if they are rotated the potential problem of shadow remains
- Vehicles are *differently illuminated* due to different day times and changing weather conditions



(a) (b)

Figure 3.9: Impact of sunshine for the training of the classifier. Two cars from the same dataset (original size 32 x 32 pixels) where the impact of sunshine becomes clear. (a) A car sample which is oriented in a western direction. The shadow in the northern direction is plain to see. (b) A car facing the northern direction. The sun is still shining and illuminates the object, however, the shadow can hardly be seen.

- Vehicles can be in *various contexts*, such as a parking area where other cars are adjacent

These problems can be countered with the following actions. In the case of the different dimensions it is necessary to split up the training data and thus also the detection procedure into several vehicle classes. It seems to be clear that it is not recommended to put large trucks in the same class as small cars. But there are also dimensional differences existent in the class of cars. If these differences are too large it is better to set up a new class. The threshold for cars being to large is related to the spatial resolution of the data and the desired precision. Of course, more classes lead to a higher manual effort when training data is prepared.

Moreover, it is not recommended to cut out samples and rotate these in order to retrieve the desired orientation and pad out the training data. This will lead to an imprecise classifier as well. The reason for this can be observed in Figure 3.9. If there is sun shine during acquisition of the images, the sun will inherently light from the south-east, south or south-west. Hence the shadow of the cars is more or less in a direction that is somewhere to the north of the object. Due to this fact shadow is a feature which becomes obvious when we compare cars of two different orientations (Figures 3.9a and 3.9b).

Furthermore, it has to be noted that a homogeneous training data set is also necessary because of a general sensitivity to noise of the chosen training procedure (Section 3.5.2). The reasonable number of training samples is determined to about 50 pieces and they are manually selected (see Section 4.3).

3.5.2 Training of the classifier

The following lines give an explanation for the training of the classifier. The explained methodology is valid for probabilistic approaches only. Every object to be examined belongs to a class. The relation to a class can be seen as a characteristic of variable Y. Based on an observation vector x, the assignment to a class should be carried out. The relationship of Y and x can be seen as a conditional probability $(Y|x)$. Furthermore,

the relation can be estimated by classifier C, which can be created, for instance, using knowledge from similar observations where the class of the object is already known.

In our case, several thousand features are calculated (Section 3.4.2) from each training patch (Section 3.5.1). Additionally, each feature consists of several elements again. Finally, these elements are concatenated to one feature vector which provides us the implicit description of the object. It is good to have many features (long feature vector), on the one hand, because the description is very detailed and could be helpful to distinguish between similar objects. On the other hand, it is problematic because it leads to a feature space of very high dimensionality.

Some training methods, for instance support vector machines (SVMs), are not suitable to handle feature spaces of such high dimensionality. The application of SVMs can be enabled when only highly informative features are selected and unimportant or redundant ones are discarded. This can be achieved, for example, by applying a principal component analysis (PCA). In contrast, the AdaBoost algorithm is inherently not limited to a maximum number of features.

Generally, Boosting creates a strong classifier by combining several weak classifiers – those that correctly classify more than 50 percent of the training data. Initially, this has been achieved by Schapire [1990] and Freund [1990], but the method was not adaptive at that time. Due to this missing characteristic, variants like AdaBoost have been developed [Freund & Schapire, 1995, 1996, 1997].

The training data consists of two classes which are car and no car. Therefore, the following description of the AdaBoost algorithm is restricted to the binary variants and does not include multi-class variants. If needed, more classes could be used to classify, for instance, trucks as well. However, it is not essential and can be done in a separate process as well.

In a first step, the training data is split up into training and evaluation parts. The evaluation part is used to determine the termination criterion. Parameters leading to termination of the iterative training procedure are the desired detection rate and the accepted false negative rate. At the beginning, all samples in the training part are treated with the same weight values. Then a binary recursive partitioning procedure generates a classification tree [Breiman et al., 1984]. Training samples which have been inaccurately classified are treated with a higher weight in the next training round. The actual goal is the special consideration of samples which are difficult to classify. In the end, all obtained trees are combined in one classifier. This can also be done in the style of a cascade [Viola & Jones, 2001].

It is necessary to add that there are two variants of training data selection for the AdaBoost algorithm. One method applies the weights directly to the training samples, and the another method uses the weights as probabilities to decide which training samples are drawn. In the following process only these training samples are used. The choice depends on the subsequent function and whether it is able to handle weighted input data [Hechenbichler, 2005]. In the following scenario, weights are directly applied to the training samples.

Moreover, it is necessary to mention that AdaBoost is quite sensitive to noise [Bauer & Kohavi, 2009]. To address this problem the gentle AdaBoost variant has been developed.

In comparison to real AdaBoost, outliers are not treated in such an extreme way. However, the best solution is still to select the training data carefully(Section 3.5.1).

Another general problem of training algorithms is termed over-fitting. It normally occurs when the training data is memorized or the noise of the training data is learned. Although over-fitting of the AdaBoost algorithm could be expected, it is seldom observed [Polikar, 2006]. A detailed explanation based of the margin theory is given by Schapire et al. [1998]. However, referring to the experience gained with car data from aerial imagery, over-fitting may happen. It occurred in situations when the group of training cars was quite inhomogeneous – in other words, noisy.

Variants of the AdaBoost algorithm are explained in the following paragraphs. The mathematical notation of the AdaBoost descriptions is similar to Hechenbichler [2005] and can be studied there in detail.

Discrete AdaBoost

The mathematical expression of the discrete AdaBoost algorithm can be written in following way:

1. Start with weightings $w_1 = ... = w_{n_L} = 1/n_L$ for the observations of training sample L.

2. Do step m as follows:

   a) The classifier $C(., L_m)$ is created using weighted observations L_m of training sample L

   b) The classifier $C(., L_m)$ is applied on L and $\epsilon_i = 1$ if the i^{th} observation is classified as false. Otherwise $\epsilon_i = 0$.

   c) The re-sampling weights are updated with $e_m = \sum_{i=1}^{n_L} w_i \epsilon_i$, $b_m = (1 - e_m)/e_m$, and $c_m = \log((1 - e_m)/e_m)$:

$$w_{i,new} = \frac{w_i b_m^{\epsilon_i}}{\sum_{j=1}^{n_L} w_j b_m^{\epsilon_j}} = \frac{w_i exp(c_m \epsilon_i)}{\sum_{j=1}^{n_L} w_j exp(c_m \epsilon_j)} \tag{3.21}$$

3. After M steps we receive the aggregated vote for the observation x:

$$\text{argmax}_j \left(\sum_{m=1}^{M} c_m I(C(x, L_m) = j) \right) \tag{3.22}$$

where e_m is a weighted sum of errors. The term c_m is the logarithmic ratio of right and wrong classified samples. The term is not arbitrarily chosen because it is responsible of the fact that half of the total weights are used for right classified samples and the other half are used for the false classified samples [Hechenbichler, 2005].

Real AdaBoost

In contrast to the discrete AdaBoost algorithm the real AdaBoost algorithm, introduced by Friedman et al. [2000], utilizes a real-valued classifying function $f(x, L)$. That means the result of $C(x, L) = 1$ is now obtained with values between the range of $]0, 1]$ and vice versa if $C(x, L) = -1$ the obtained confidence values range from $[-1, 0[$. The class indicator Y is now defined $Y \in \{-1, 1\}$ and the algorithm is mathematically descried as follows:

1. Start with weightings $w_1 = ... = w_{n_L} = 1/n_L$ for the observations of training sample L.
2. Do step m as follows:

   a) The classifier $C(., L_m)$ is created using weighted observations L_m of training sample L

   b) The classifier $C(., L_m)$ is applied on L and $p(x_i) = \hat{P}(\tilde{Y}_i = 1|x_i)$ is received

   c) Based on these probabilities a real-valued classifier is developed

$$f(x_i, L_m) = 0.5 * log \frac{p(x_i)}{1 - p(x_i)} \tag{3.23}$$

   and the weights are updated for the next step:

$$w_{i,new} = \frac{w_i exp(-\tilde{Y}_i f(x_i, L_m))}{\sum_{j=1}^{n_L} w_j exp(-\tilde{Y}_j f(x_j, L_m))} \tag{3.24}$$

3. After M steps we receive the aggregated vote for the observation x:

$$sign\left(\sum_{m=1}^{M} f(x, L_m)\right) \tag{3.25}$$

Gentle AdaBoost

The gentle AdaBoost algorithm proposed by Friedman et al. [2000] introduces a new term for updating the weights. Instead of the logarithmic quotient of the probabilities (Equation 3.23) the new term is calculating the difference of the two probabilities (Equation 3.26). The goal is to put less emphasis on outliers. Furthermore, it is numerically more stable and there is no need to close gaps in the definition of the function [Hechenbichler, 2005].

The mathematical description is as follows:

1. Start with weightings $w_1 = ... = w_{n_L} = 1/n_L$ for the observations of training sample L.

2. Do step m as follows:

   a) The classifier $C(., L_m)$ is created using weighted observations L_m of training sample L

   b) The classifier $C(., L_m)$ is applied on L and $p(x_i) = \hat{P}(\tilde{Y}_i = 1|x_i)$ is received

   c) Based on these probabilities a real-valued classifier is developed

$$f(x_i, L_m) = p(x_i) - (1 - p(x_i)) \tag{3.26}$$

   and the weights are updated for the next step:

$$w_{i,new} = \frac{w_i exp(-\tilde{Y}_i f(x_i, L_m))}{\sum_{j=1}^{n_L} w_j exp(-\tilde{Y}_j f(x_j, L_m))} \tag{3.27}$$

3. After M steps we receive the aggregated vote for the observation x:

$$sign\left(\sum_{m=1}^{M} f(x, L_m)\right) \tag{3.28}$$

In some empirical tests gentle AdaBoost leads to better results than real AdaBoost [Lienhart et al., 2003]. However, in our situation it is not superior to real AdaBoost. Empirical tests showed a slightly better result when using the real AdaBoost variant. Hence, the results in Chapter 5 are based on the real AdaBoost algorithm.

3.5.3 Vehicle classification

The obtained classifier from Section 3.5.2 is now applied to the remaining parts of the image. Remaining parts are those areas which have neither excluded by the disparity image (Section 3.2.1) nor by the segmentation procedure (Section 3.3).

Resulting classifiers consist of several cascades. The final number of cascades depends on the termination criterion which was set at the beginning. Training data, for instance, which consist of very different positive samples and negative samples lead to fewer cascades. Moreover, the number of utilized weak classifiers per cascade increases when used later in the process. This can be seen in Figure 3.10 where utilized features are marked in different colors. The colors are related to the stage of the cascade. The demonstrated classifier consists of three stages. The first stage (orange) has features with a larger area as it is usual.

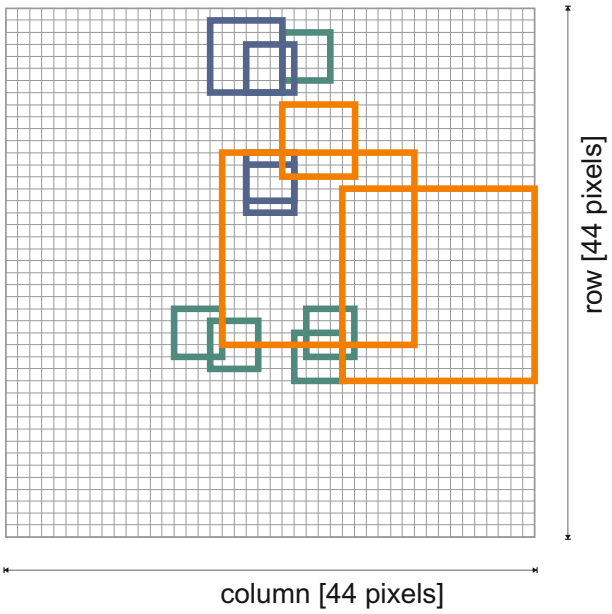


Figure 3.10: Example of what kind of features are used for the classifier. The classifier was trained with cars having vertical orientation. First stage of the classifier uses features colored orange, while second and third stage are blue and green, respectively. The result can be compared to Figure 3.5 in order to visually estimate which significant edges have been used.

3.6 Final weighted selection of vehicles and coordinate transformation

The two finalizing steps take care about multi detections and the global coordinates. The first point is necessary to receive an optimal detection result, whereas the second point is used to transform the data in a sharable format.

3.6.1 Final weighted selection of vehicles

Finally, after the steps described above remains a number of multiple detections. This phenomenon can be explained by the inhomogeneity of the training data (Section 3.5.1).

A standard solution is the use of the mean shift algorithm [Fukunaga & Hostetler, 1975]. However, the presented approach takes a different path and uses a faster non-iterative technique. All confidence values of the detections are summed up in a Gaussian weighted manner according to their distance within a certain area. That means values far away, but still in a certain area, contribute less. At the end of the process the candidate with the highest votes within the limited area is accepted and proven to be the real center of the detected car.

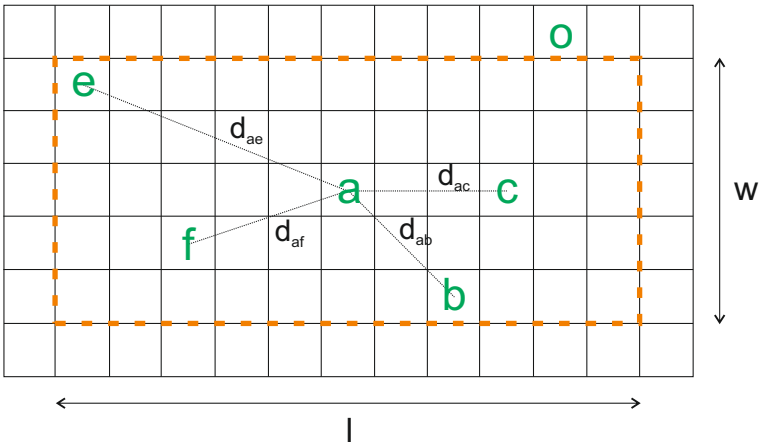


Figure 3.11: Sketch showing how multi detections are treated. The grid is a small part of the search image. The cell size of the sketch is equivalent to the pixel size of the original image. The green variables stand for confidence values. l, w represent length and width of the rectangle set to the average size of a small car. d_{xx} is the Euclidean distance between two variables. Variable o is outside the rectangle and therefore ignored – in the same manner as all other variables outside.

A graphical explanation is shown in Figure 3.11. The green variables are confidence values voting for a car. The confidence value of variable a is extended by surrounding variables which lead to a new vote a^*:

$$a^* = a + g(d_{ab})b + g(d_{ac})c + g(d_{ae})e + g(d_{af})f \tag{3.29}$$

where surrounding variables b, c, e, f are weighted according to their distances d_{xx}. The Gaussian weighting procedure is symbolized by function $g(d_{xx})$. Variable o in the figure is outside the considered area and is therefore ignored. It could be a completely different car due to the larger distance to a.

The fixed certain area usually is the size of a small average car. The size can be set because the GSD of the aerial image is known. Moreover, because of the fixed certain area also false positives can be identified and rejected. A drawback is the rotation invariance, but this prerequisite has already been set in previous steps of the strategy.

3.6.2 Transformation of vehicle positions to global coordinates

After all preceding steps, the current position of the cars is only defined by local coordinates which are consisting of row and column in the search image. In order to provide the extracted traffic information in a useful format for further-processing partners, global coordinates are requested. Therefore, Gauss-Krueger or Universal Transverse Mercator (UTM) coordinate systems are suitable.

Since the search image is not ortho-rectified and the geocode is missing a transformation is needed. Here again the colinearity equation is used.

$$X = X_0 + (Z - Z_0)\left[\frac{r_{11}(x' - x'_0) + r_{21}(y' - y'_0) - r_{31}c}{r_{13}(x' - x'_0) + r_{23}(y' - y'_0) - r_{33}c}\right] \tag{3.30}$$

$$Y = Y_0 + (Z - Z_0)\left[\frac{r_{12}(x' - x'_0) + r_{22}(y' - y'_0) - r_{32}c}{r_{13}(x' - x'_0) + r_{23}(y' - y'_0) - r_{33}c}\right] \tag{3.31}$$

where X, Y, Z are the coordinates of the object (vehicle), X_0, Y_0, Z_0 are the coordinates of the projection center in the coordinate system of the object. The principal point is defined by x'_0, y'_0 and the position in the image is given by x', y'. The calibrated focal length is involved by c.

The height of the projection center Z_0 can be taken from the GPS receiver of the aircraft and the corresponding height of the object is taken from the global DEM (e.g., SRTM).

For detailed information about the UTM coordinate system and the further transformation starting from the Cartesian coordinates X,Y,Z you are referred to Kahmen [2005].

3.7 Car candidate validation using background and color information

The color information of the vehicle on the one hand and the background of the vehicle on the other hand are features that have been utilized quite seldom for car detection (Section 2.2). However, the potential of these features is examined and thus an approach to validate vehicle candidates is developed. The approach is mainly inspired by Heitz & Koller [2008] and Chang & Krumm [1999].

The idea is based on the development of a technique which has different roots than normal segmentation or classification techniques. The reason is that the method is aimed to apply as validation technique and no true positives should be removed. A SVM or another machine learning algorithm would need a slack variable to do so.

Firstly, CCHs are calculated from training data but these are not given to a machine learning technique. Instead, the distribution is calculated and generalized using a beta function. The idea behind is the higher transparency and also the better understanding and the intervention possibilities. The corresponding workflow of the approach is shown in Figure 3.12. The work is also published by Leister et al. [2013]. A detailed description plus experiments can be found in Leister [2013].

3.7.1 Background separation and HSV color space

The way to obtain the background of the vehicle candidate and the utilized color space are described in the following subsections.

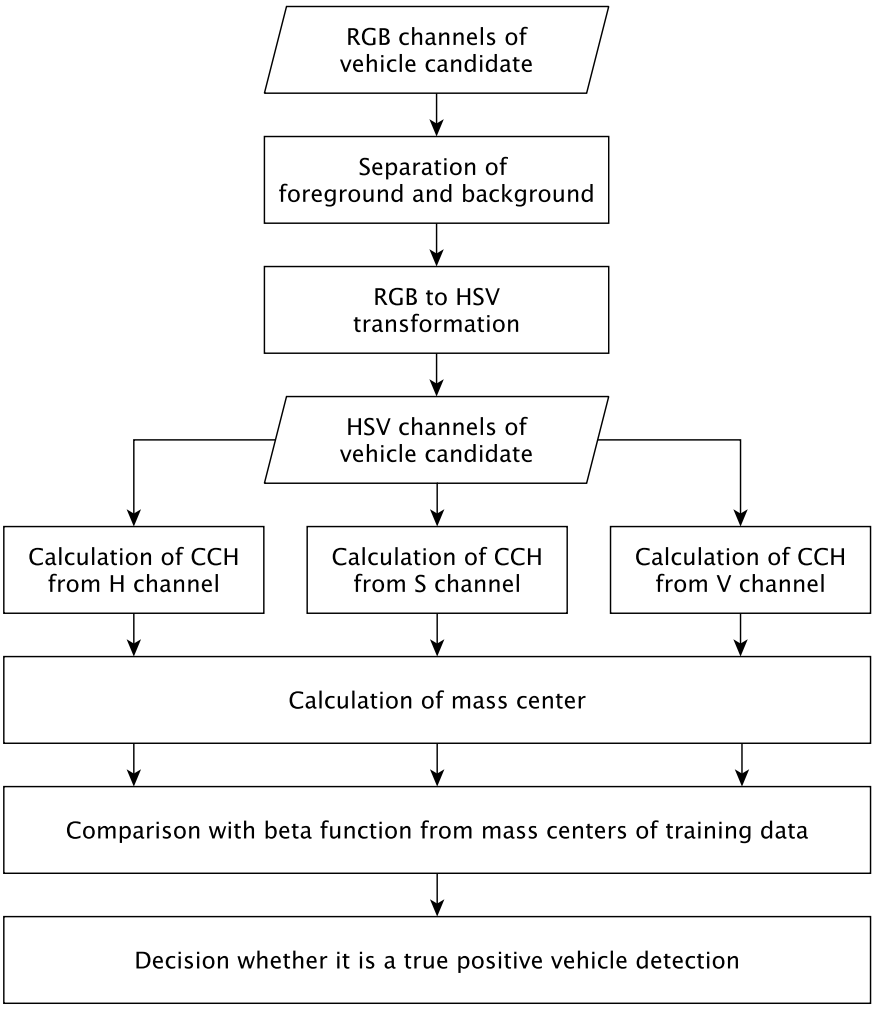


Figure 3.12: Workflow of the vehicle validation technique.

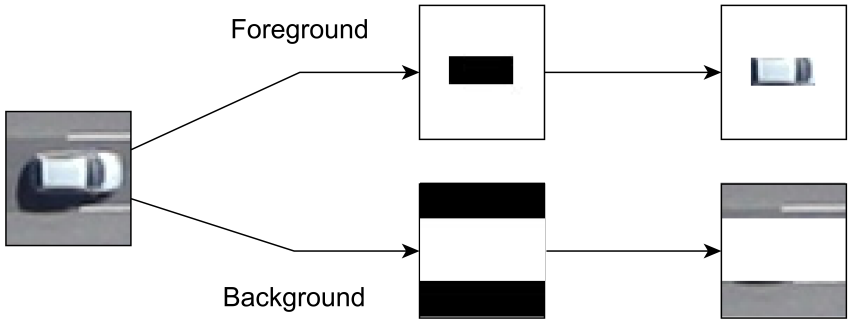


Figure 3.13: Extraction of foreground for validation purpose. Foreground (potential car candidate) and background (potential street) can be set as a fixed area since the orientation of the cars is assumed to be known. The foreground is in the center of the patch while the background is restricted to the areas to the left and right of the car.

Background separation

Firstly, foreground and background of each car candidate are examined separately. However, the technique is identical for foreground and background areas. At the beginning, all candidates are represented in the RGB color space.

In order to apply a mask the orientation of the vehicles has to be known. Since the presented approach is planned to act as validation method, the orientation can be obtained from the preceding detection algorithm. Alternatively, road databases can be used to determine the potential driving direction of the cars. Additionally, it is assumed that a car is in the center of the examined image patch.

The principle of the fore- and background mask is depicted in Figure 3.13. The size of the foreground mask is determined by the average size of a car. It is derived from the dimensions of 30 training cars. However, pixels close to the contour of the cars have been ignored. The reason is that often artifacts occur at these positions due to shadow. The major objective of the presented method is to get statistical information about color and lightness of the car and the background, and thus artifacts could adulterate the statistics.

The background area is represented by the remaining rectangular areas to the left and the right side of the cars. The area in front and behind a car is not used in the further process because often cars are parked in a row and other cars which disturb the process can be found at these positions.

HSV Color Space

After the separation a transformation into the HSV color space is performed. From that moment on the color information and the intensity can be independently accessed. A special property of the HSV color space is the necessity of only one channel to define the color value. The transformation from RGB to HSV color space can be found in Gonzalez & Woods [2007].

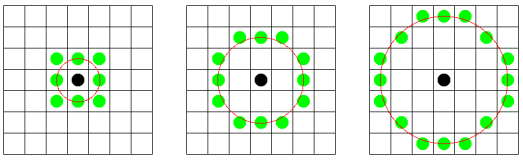


Figure 3.14: CCH and example of a circular symmetric structure of neighborhood. This variant is utilized.

3.7.2 CCH feature and likelihood calculation

Color co-occurrence histogram

Subsequently, color co-occurrence histograms (CCH) are calculated. Co-occurrence histograms are based on the relation of neighboring pixels and give a statement about the properties of the texture [Liu et al., 2012]. A graphical explanation can be seen in Figure 3.14.

The calculation of a CCH can be mathematical written as:

$$H(I,d) = \sum_x \sum_y \begin{cases} 1 & \mid I(p_x, p_y) = I(x,y) \cap d = 1 \\ 0 & \text{else} \end{cases} \tag{3.32}$$

The factor $d = 1$ has been chosen because experiments have not shown significant differences between $d = [1, 3, 5, 10]$. On the other hand with $d = 1$ the fewest comparisons need to be done and calculation time can be saved.

Likelihood decision

A closer empirical examination of the histograms shows that often an accumulation around a certain value occurs [Leister et al., 2013]. Hence, the implication is that the mean of the histogram is able to provide enough information for the following decision. To this end, the mean of a CCH is calculated as shown in Equation 3.33.

$$m = \frac{1}{\sum_I h(I)} \sum_I h(I) \cdot I \tag{3.33}$$

where $h(I)$ is the I-th value in the CCH.

A training set consisting of samples from all used classes is necessary for the following process. The reference data is essential for a correct classification of the candidates and the possession of an appropriate large dataset of reference data is recommended, in order to be able to make a significant statement. Calculating the mean values of these training candidates leads to characteristic histograms for all three categories. Finally, we obtain three CCHs from every candidate and out of it three means (m_H, m_S, m_V). Subsequently,

every value is compared with the values of the corresponding histogram of the three classes which we calculated from the training data. For example, the process is as follows. m_{H} is compared to the values of the hue-histogram of cars, roads and vegetation. We take the three corresponding (i. e. $m_{\mathrm{H}} \rightarrow [h_{\mathrm{car}}(m_{\mathrm{H}}), h_{\mathrm{str}}(m_{\mathrm{H}}), h_{\mathrm{veg}}(m_{\mathrm{H}})]$) values $h_{\mathrm{cat}}(m)$ and compare them with each other. The nine quantities ($q_{\mathrm{H,car}}$, $q_{\mathrm{H,str}}$, …, $q_{\mathrm{V,veg}}$) stating to which distribution the mean value of a candidate belongs to are calculated using Equation 3.34.

$$q_{\mathrm{chan,cat}}(m_{\mathrm{chan}}) = \frac{h_{\mathrm{chan,cat}}(m_{\mathrm{chan}})}{\sum_{\mathrm{cat}} h_{\mathrm{chan,cat}}(m_{\mathrm{chan}})} \tag{3.34}$$

In the next step we multiply the quantities of the same category to get a combined value (Equation 3.35).

$$k_{\mathrm{cat}} = \prod_{\mathrm{chan}} q_{\mathrm{chan,cat}}(m_{\mathrm{chan}}) \tag{3.35}$$

This gives us three values, named $k_{\mathrm{car}}, k_{\mathrm{str}}$ and k_{veg}, describing the frequencies for the examined area of being car, road or vegetation. The new values k_{cat} are then assumed to be directly correlated to the likelihood of being such a candidate.

Based on these six (2×3) k_{cat} values of the foreground and the background, a decision can be made whether a candidate is a car or not. We can compare these values to each other and find scenarios which mostly show cars on streets, pure streets or others.

The following three rules describe conditions when the candidate is supposed to belong to the no car class:

- Road in foreground:

  $[k_{\mathrm{str}}(\mathrm{foreground}) > k_{\mathrm{veg}}(\mathrm{foreground})] \wedge [k_{\mathrm{str}}(\mathrm{foreground}) > k_{\mathrm{car}}(\mathrm{foreground})]$

- Vegetation in background:

  $[k_{\mathrm{veg}}(\mathrm{background}) > k_{\mathrm{str}}(\mathrm{background})] \wedge [k_{\mathrm{veg}}(\mathrm{background}) > k_{\mathrm{car}}(\mathrm{background})]$

- A too small difference between foreground and background:

  $\sum_{\mathrm{chan}} |m_{\mathrm{chan}}(\mathrm{foreground}) - m_{\mathrm{chan}}(\mathrm{background})| \leq \mathrm{threshold}$

The threshold can be specified dependent on different light conditions and sensor properties. In our experiments, the threshold ranged from 10 to 15. When the difference was too low, foreground and background were the same category.

Instead of using the histograms of the training data, the corresponding beta-functions give us the possibility to estimate the values of the histograms $h_{\mathrm{chan,cat}}$ where we do not have training data.

3.8 Moving-object incorporation

The entire approach presented is designed to detect parked or 'not moving' cars. However, often there are some cars moving in a scene and this incidence can be used to enhance the overall detection quality – also of parked cars.

Generally, moving cars are easier to detect than parked ones. The reason is that for a detection of these objects the search space can be reduced with simple methods (as shown below). The idea is to detect these moving cars first and combine the results with the strategy described in the previous sections. The benefit is that, besides all possible complications, at least all moving cars are reliably detected. Additionally, it can be assumed that no other detection is valid within a certain radius of the reliable detection. This helps to identify false positive detections from a less reliable detection procedure such as the procedure for stationary cars.

Firstly, a color space is chosen which is technically oriented. That means per definition the color space must be a linear transformation of the RGB color space. The utilized color space is $I1I2I3$ and can be calculated from RGB color space in the following way (Equation 3.36).

$$\begin{bmatrix} I1 \\ I2 \\ I3 \end{bmatrix} = \begin{bmatrix} 1/3 & 1/3 & 1/3 \\ 1/2 & 0 & -1/2 \\ -1/4 & 1/2 & -1/4 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3.36}$$

where R, G, B represent the red, green, blue channels and $I1$, $I2$, $I3$ are the resulting channels of the $I1I2I3$ color space model.

Furthermore, three subsequent images are co-registered (e.g., using SIFT) or, if available, the geocode of the images is used. Subsequently, the difference of the current image and the previous image, and the difference of the current image and the subsequent image are calculated. The two resulting difference images are linked with the Boolean AND. The method expressed in formulas can be seen in Equation 3.37 where the first difference image D_1 is calculated [Rehrmann & Birkhoff, 1995]:

$$D_1(t_1, t_2, x, y) = \begin{cases} 1, \text{ if } & |I_{I1}(t_2, x, y) - I_{I1}(t_1, x, y)| \\ & + |I_{I2}(t_2, x, y) - I_{I2}(t_1, x, y)| \\ & + |I_{I3}(t_2, x, y) - I_{I3}(t_1, x, y)| \; > d_{min} \\ 0, \text{ else} \end{cases} \tag{3.37}$$

where the functions of the images are $I_{I1}(t, x, y)$, $I_{I2}(t, x, y)$ and $I_{I3}(t, x, y)$. The parameter t is the triggering time whereas x and y are row and column coordinate of the three different channels $I1$, $I2$, $I3$. The parameter d_{min} is a threshold which is necessary for excluding intensity changes of pixels due to sensor noise, various illuminations or the different illustration geometry. Afterwards, those two calculated difference images are linked (Equation 3.38):

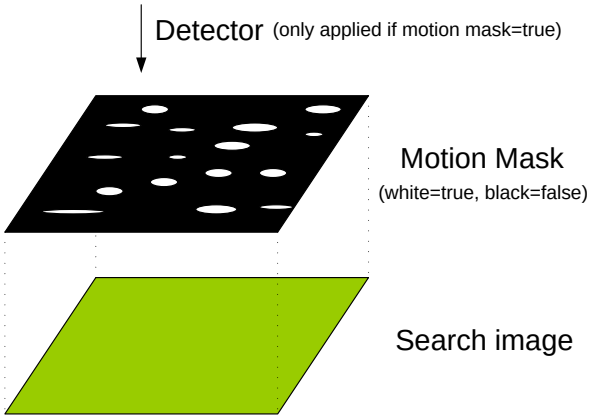


Figure 3.15: Schematically explanation of the utilized motion mask: The binary motion mask (black and white layer in the Figure) is calculated according to Equation 3.38. Black and white indicate areas where stationary and moving objects are expected, respectively. Furthermore, the motion mask is projected to the search image. The detector, which is schematically shown in the Figure as arrow perpendicular to the motion mask, is applied to the search image. However, the detector is only applied to areas where the motion mask is set to true (white areas). Black areas of the motion mask are skipped.

$$D_2(t_1,t_2,t_3,x,y) = \begin{cases} 1, \text{ if} & D_1(t_1,t_2,x,y) = 1 \\ & \wedge\, D_1(t_2,t_3,x,y) = 1 \\ 0, \text{ else} & \end{cases} \tag{3.38}$$

with $D_1(t_1,t_2,x,y)$ difference image of previous and current image and $D_1(t_2,t_3,x,y)$ difference image of current and consecutive image.

The utilized technique is standard in the field of moving object detection from video data where very high imaging frequencies provide more information due to the high sampling rate. In the field of lower sampled information it works up to a certain limit. Prerequisites are at least three images of the same area. This situation can be also expected for low-frequent aerial images due to their usually large field of view and the resulting overlap.

Finally, the obtained binary mask $D2$ is used to identify candidate areas of moving objects. These areas are further examined using the detector from Section 3.5.3. The principle is schematically explained in Figure 3.15. A fusion of detected moving cars with non-moving cars detected by the previously explained main technique (Section 3.1–3.5.3) is carried out with the method explained in Section 3.6.1. There, vehicle candidates from the approach for moving cars are introduced with higher weights.

4 Experiments

The following sections describe the data which is used to test the car extraction technique explained in Chapter 3. Moreover, also the conduction of the experiments is described. Results of the experiments are then shown in the subsequent Chapter 5.

4.1 Sensors and platforms

The general topic of this thesis is car extraction from aerial imagery. A fundamental part of this process is the nature of the image data. Therefore, in the following text properties of the used sensors including their optimal settings are examined. Utilized airborne sensors are on the one hand, a low cost camera system named 3K camera system and its successor the 3K+ camera system. On the other hand, a professional photogrammetric camera system named UltraCam Eagle is used. All of these systems are specially constructed for the use in aircraft.

Aircraft used for the 3K/3K+ camera system are the Cessna 208B Grand Caravan or the Dornier Do 228-212. The carrier of the UltraCam is unknown.

The experiments concerning the camera settings and image quality are only carried out with the 3K/3K+ camera systems. Reason for this is the lack of possible flight opportunities with the UltraCam sensor (Section 4.1.2).

4.1.1 3K and 3K+ camera systems

The 3K/3K+ camera systems have been developed for the purpose of mapping and traffic monitoring large areas. Each of them is composed of 3 off-the-shelf cameras namely Canon EOS 1Ds Mark II and Canon EOS 1Ds Mark III for 3K and 3K+, respectively. A picture of the 3K+ camera system is shown in Figure 4.1. Detailed information of the camera systems can be taken from Table 4.1.

The 3K/3K+ camera systems generally have fewer technical equipment features than professional aerial camera systems e.g., no motion forward control. Also the ground sampling distance is lower, mainly due to different lenses. However, besides the significant lower price, there is an important advantage of the 3K/3K+ camera system concerning the frame-rate. In case of the car extraction strategy presented here a higher frame rate is useful for the calculation of the disparity map (Section 3.2.1) and for the optional moving object incorporation (Section 3.8). The calculated disparity map will be more accurate and reliable by using redundant information of overlapping images. Also the moving

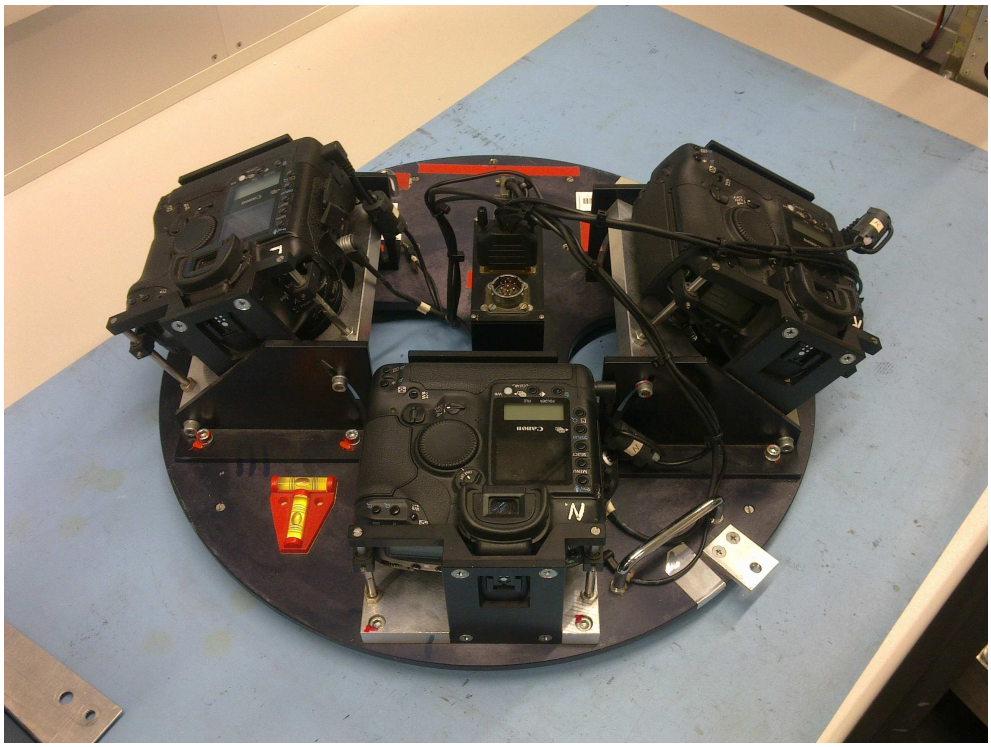


Figure 4.1: The 3K+ camera system; unmounted in the laboratory.

Table 4.1: Specification of 3K and 3K+ camera systems.

	3K	3K+
Cameras	3 × EOS 1Ds Mark II	3 × EOS 1Ds Mark III
Sensor	36 × 24 mm CMOS	36 × 24 mm CMOS
Physical pixel size	7.21 μm	6.41 μm
Image size	3 × 4992 × 3328 (16.7 MPix)	3 × 5616 × 3744 (21.0 MPix)
Max. frame rate	3 Hz [a)]	5 Hz [b)]
File size	20 MByte (RAW)	25 MByte (RAW)
Aperture	1.4 – 22	1.4 – 22
Shutter speed	1/8000 – 30 s	1/8000 – 30 s
Lenses	Canon EF 1.4/50 mm	Zeiss Makro-Planar 2/50 mm
GSD [c)]	15 cm	13 cm
Data rate	8.3 MByte/s	9.8 MByte/s
FMC	no	no

[a)] only up to 50 images
[b)] only up to 63 images
[c)] at a flight altitude of 1000 m

object detection can be simplified and speeded up by a higher frame-rate. Additionally, applications which are interested in traffic flows and vehicle velocities can extract the desired information of temporal change more precisely.

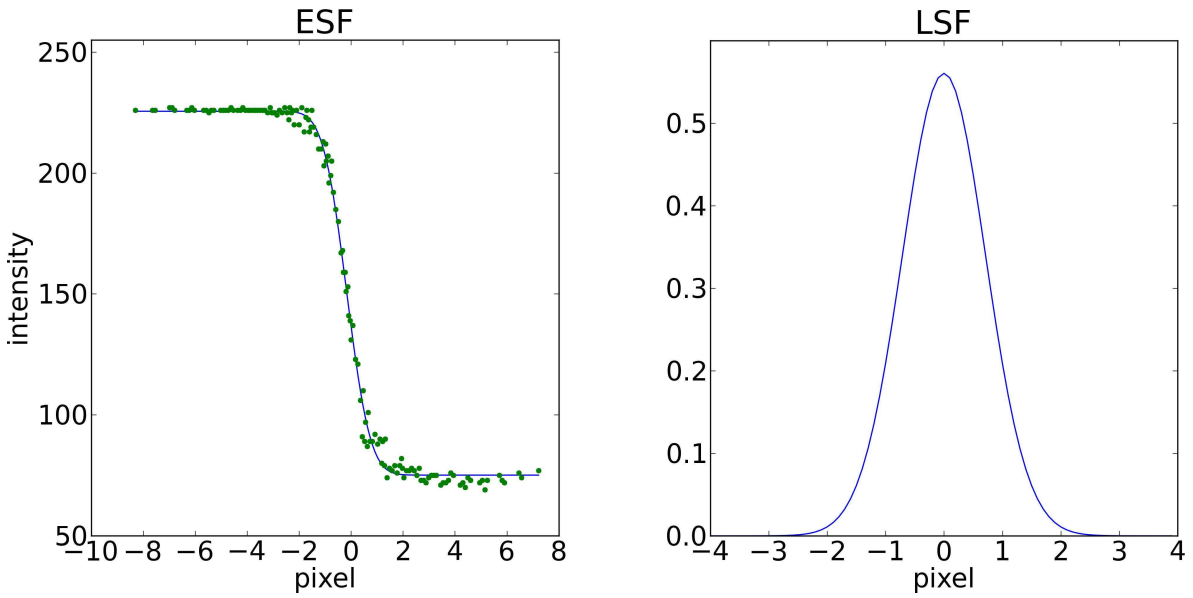
Important facts for information extraction from aerial imagery are the radiometric properties. Because this information is directly responsible for the image quality. Based on that knowledge, general statements about sharp edges or objects specific reflections are possible.

The following paragraphs give information about the radiometric properties of the 3K/3K+ camera system. The goal is to optimize the use of off-the-shelf cameras for airborne monitoring purposes, i.e., to acquire images with best resolution and contrast in the presence of forward motion blurring and changing incoming radiance. In contrast to high level photogrammetric systems, the forward motion blurring of off-the-shelf cameras is reduced by short exposure times, which worsens the conditions for achieving radiometrically optimal images. As the internal processing of the camera has no changeable parameters, it works like a black box and there is no further influence on how they affect the radiometric quality. The remaining free configurable parameters are the f-number and the ISO speed which are dependent on each other, so that only an appropriate combination allows the best possible imaging result. Concise information about the influence of these parameters on the radiometric performance is given in the following paragraphs.

The f-number is the focal length divided by the 'effective' aperture diameter. A low fnumber (e.g., 2.0) passes a lot of light to the sensor but also results in blurring due to the larger circle of confusion. However, the image sharpness in the focal plane varies with the relative aperture size. Additionally, there is optical vignetting which is sensitive to the f-number and lens architecture. In general, the blurring can be cured by a reduction in aperture of 2 steps. Due to the lens properties of 3K/3K+ f-numbers greater than 4.0 are able to produce satisfying results. For instance, Zeiss Makro Planar 2 has an aperture range of f/2 to f/22.

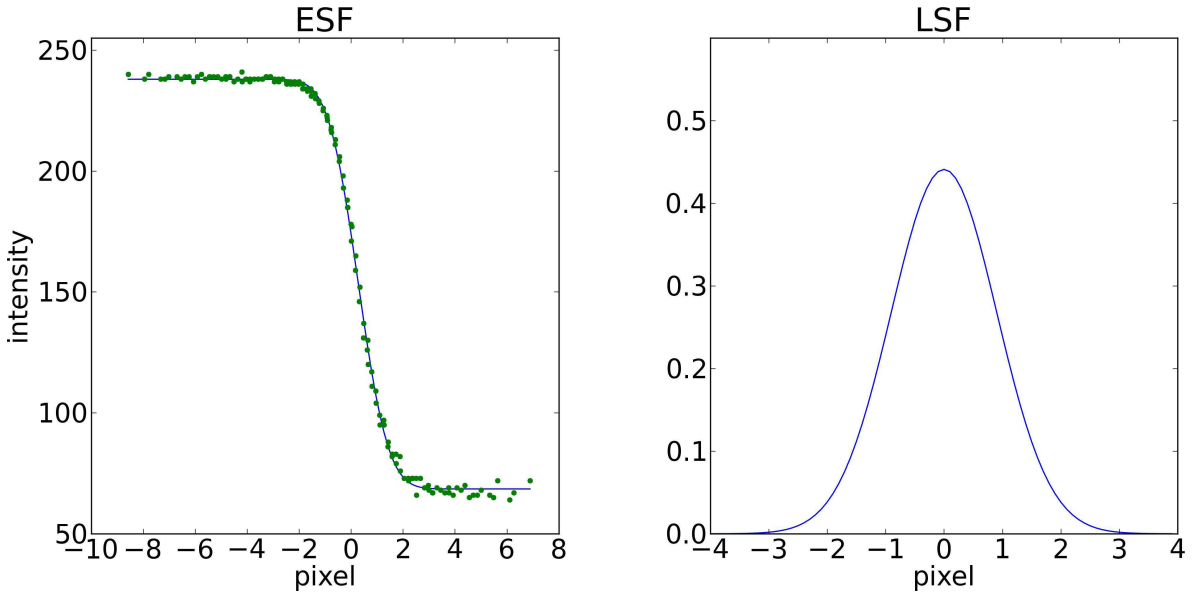
The shutter speed is indirectly proportional to the light reaching the sensor. As mentioned, short exposure times are aspired in order to reduce forward motion blurring. A flying velocity of e.g., 70 m/s at 1000 m altitude, with a shutter speed of 1/2000 s results in 3.5 cm movement which approximately corresponds to 1/4 pixel. Higher shutter speed values reduce the incoming light and thus enforce the f-number and film speed to inappropriate values. Our test supports the assumption that a shutter speed of 1/2000 s is an acceptable compromise. Edge spread functions (ESF) and their corresponding line spread functions (LSF) based on an image with shutter speed 1/2000 s are shown in Figure 4.2 and one with shutter speed 1/8000 s in Figure 4.3. Comparing both LSFs shows that the image with lower shutter speed has sharper edges (sigma 0.73, respectively sigma 0.92). Obviously, the reason is that a faster shutter speed is not able to compensate the lower f-number.

ISO speed is the measure of the sensor's sensitivity to light. Higher values result in noisy images. Hence, the aim is a low ISO speed, but this can be an impossible requirement – especially on cloudy days. The impact of different ISO speed parameters can be observed in Figure 4.4, which illustrates that higher ISO values cause noisy images. A test campaign with the 3K+ sensor was performed with different f-numbers, shutter speed values and ISO settings to find out the best camera settings with the highest effective GSD. The range



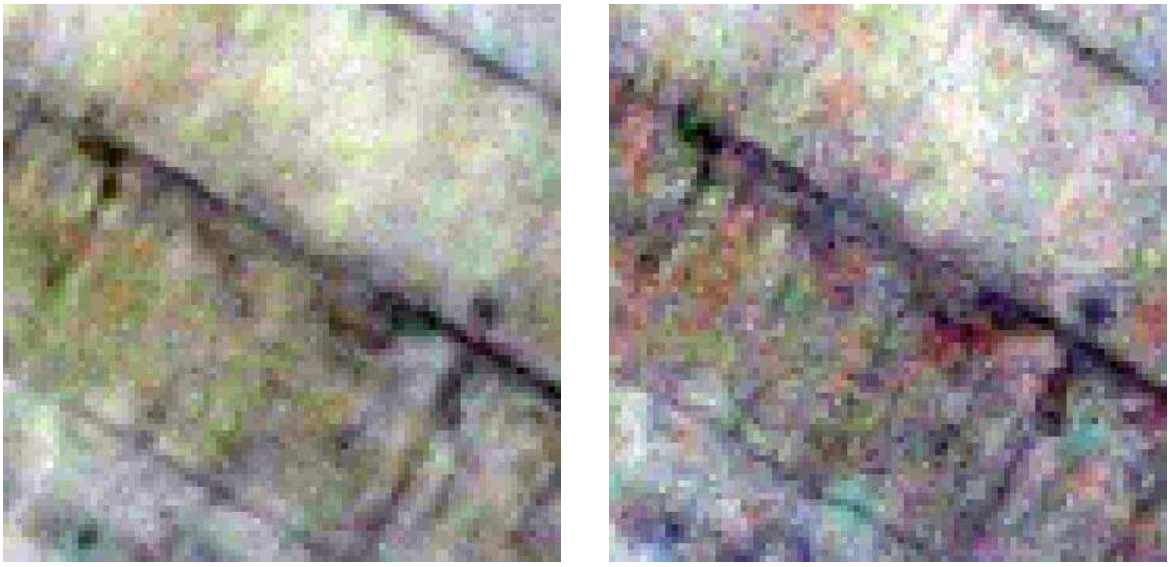
(a) (b)

Figure 4.2: Edge spread function (a) and line spread function (b) of an image from the 3K+ camera system with 1/2000 s exposure time.



(a) (b)

Figure 4.3: Edge spread function (a) and line spread function (b) of an image from the 3K+ camera system with 1/8000 s exposure time.



(a) (b)

Figure 4.4: Images taken with two different ISO speed settings. Visualization of 3K+ sensor noise (enlarged areas of the concrete areas shown in the image of Figure 4.5b). The images have been taken with following settings: (a) ISO 250, 1/2000 s, f2.8 (b) ISO 1600, 1/2000 s, f5.6.

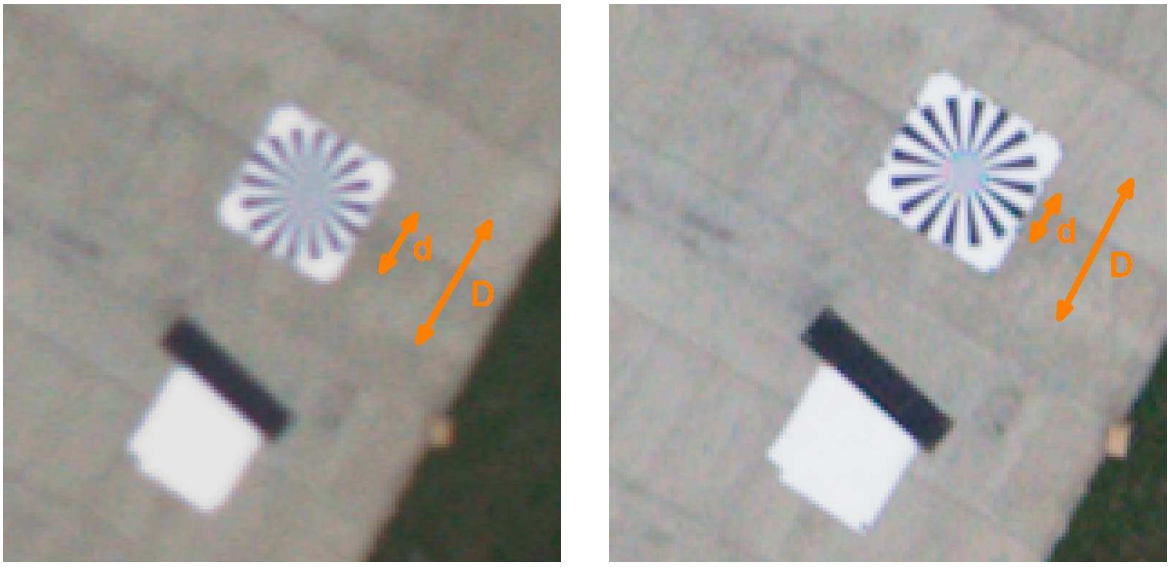
of settings for the shutter speed was 1/1000 s to 1/8000 s, for the f-numbers ranging from 2.8 to 5.6, and for the ISO values from 250 to 1600. The campaign showed that the best results are obtained by taking fixed settings for the exposure time and the f-number, while the ISO setting is variable according to the illumination conditions. The settings vary from case to case, for instance areas covered by concrete have a higher reflectance than forests. The resolution (effective GSD) of the 3K/3K+ camera system was determined by a Siemens star with a diameter of five meters as seen in Figure 4.5. The formula to obtain the effective GSD l using a Siemens star is depicted in Equation 4.1.

$$l = \frac{\pi \cdot d}{n} \tag{4.1}$$

where d is the diameter of the blurred area in the center of the Siemens star and n is the number of black and white bars.

According to that experiment we obtain an effective GSD of 18.6 cm for the 3K and 13.2 cm for the 3K+ camera compared to the theoretical GSD of 15 cm respectively 13 cm from 1000 m above ground. Also the signalized edge is sharper in the 3K+ image than in the 3K image. The standard deviation of the LSF is $\sigma = 1.07$ pixels for the 3K image and $\sigma = 0.69$ pixels for the 3K+ one.

For further detailed information about the 3K+ camera system refer to Kurz et al. [2012].



(a) (b)

Figure 4.5: Image of Siemens star and black/white edge; used to measure resolution and line spread function of (a) 3K (b) 3K+ camera systems. The orange d depicts the blurred inner circle and the orange D depicts the entire diameter of the Siemens star.

4.1.2 UltraCam Eagle camera system

Although the car extraction strategy presented here has been developed with regard to low-cost camera systems like the 3K/3K+ ones, it should be shown how generally applicable the technique is. Therefore, an alternative sensor is introduced which is the UltraCam Eagle from Microsoft (Table 4.2). It has more technical features such as forward motion compensation. Additionally, a higher resolution is achievable compared to 3K/3K+ at the same flying altitude which is due to the 80 mm lens and the larger sensor area. However, the frame rate is much lower compared to 3K/3K+. Nevertheless, there is still a sufficient overlapping area which enables us to calculate the disparity image (Section 3.2.1).

There is no detailed examination of the radiometric properties because the camera itself has never been in my possession. However, there is a project of the German Society for Photogrammetry, Remote Sensing and Geoinformation which dealt with this issue [Cramer, 2010; von Schönermark, 2010]. The data have been kindly made available by the Bavarian State Office for Surveying and Geoinformation.

4.2 Data and scenes

The following scenes have been utilized for carrying out the experiments described in Section 4.3. Moreover, the results based on these images are then presented in Chapter 5. Furthermore, Table 4.3 gives an overview of the main properties of the test scenes.

Table 4.2: Specification of the UltraCam Eagle camera system.

	Microsoft UltraCam
Camera	UltraCam Eagle (panchromatic) [color]
Sensor	104.05 × 68.02 mm CCD
Physical pixel size	5.2 μm
Image size [pixel]	20010 × 13080 (260 MPix) [3 × 6670 × 4360]
Max. frame rate	0.56 Hz
File size	842 MByte
Aperture	5.6
Shutter speed	1/500 – 1/32 s
Lenses	80 mm
GSD [a)]	6.5 cm
Data rate	462.5 MByte/s
FMC	yes (max. 50 pixels)

[a)] at a flight altitude of 1000 m

Table 4.3: Main properties of the test scenes. All scenes are from the inner city area of Munich, Germany.

Dataset No.	Pixels	Hz	GSD	View	Date (local time)	Sensor
1	677 × 268	3	13 cm	nadir	06-07-2011 11:51	3K+
2	799 × 288	3	13 cm	nadir	06-07-2011 11:51	3K+
3	720 × 691	3	13 cm	nadir	06-07-2012 12:08	3K+
4	1752 × 520	3	13 cm	nadir	26-04-2012 11:08	3K+
5	1632 × 474	0.56	20 cm	nadir	26-05-2012 - -:- -	UltraCam

Datasets of the 3K+ camera system are only taken by the center camera of the system and thus provide nadir view. Alternatively, images from the left and the right camera would provide an oblique view which leads to more occluded areas – especially in urban areas with high buildings.

4.2.1 Dataset 1 - 3K+, small road, city center, Munich

Dataset 1 is a small road which is oriented in a west-east direction and located close to the city center of Munich. The dataset has been chosen because of the typical inner city structure – high buildings are to the left and the right side of the road. These kinds of image are not easy to handle for unsophisticated car detectors due to car like structures on the roofs and on the facąades which can cause false positive detections. Additionally, parked cars are very close to facąades and roofs which is useful to test the accuracy of the ground extraction step.

The dataset is of further interest because often the accuracy of road databases is very low in areas where small roads are bordered by high houses. The reason is mainly the low availability of GNSS satellites due to shadowing effects. When a road database is set up, which is usually done by GNSS receiver equipped vehicles, the resulting inaccuracy of the measuring points has to be accepted

4.2.2 Dataset 2 - 3K+, small road, city center, Munich

Dataset 2 is similar to Dataset 1 and should be used to confirm the results achieved with Dataset 1. Here, the road is also bordered by houses, and cars are also close to the roof (aerial view). Additionally, at the time when the image was taken a garbage disposal is at work. This introduces two interesting objects into the image. A waste container which is sometimes classified as a car due to its rectangular shape, and the garbage collection truck which should not be mistakenly classified as a car because it is a truck, not a car. As further features vegetation and shadow areas are included. Shadow areas can pose problems due to the low contrast between car and background. Moreover, texture of vegetation sometimes leads to errors

4.2.3 Dataset 3 - 3K+, big road, inner-ring road, Munich

Dataset 3 is from the inner-ring road of Munich which is between the densely populated inner city area and the sparsely populated areas some kilometers off the center. At the time the image was taken, there was a large construction site in this area to build a tunnel. This results in the road course changing very quickly (sometimes daily) and road databases can no longer be used. Furthermore, the scene is interesting due to the slight curve which can be an indicator of how stable the detection algorithm is related to the orientation of the cars. Additionally, many road markings are present which can be used to test whether the detection is affected by these objects. Last but not least, the surrounding of the road is lower compared to the inner city areas (Dataset 1+2), hence the performance of the ground extraction step can be easily evaluated for this situation as well.

4.2.4 Dataset 4 - 3K+, TUM, Arcisstrasse, Munich

Dataset 4 is from the road passing the main entrance of the Technical University in a north-south direction. The example scene is marked with a yellow rectangle in the original scene in Figure 4.6. However, just part of the original scene is used to focus the discussion more on details. Several interesting objects are in this image, ranging from debris containers in the parking lane to bike paths which are painted on the road – both can cause false positives. Additionally, many cars are partly occluded due to vegetation, and many cars are partly or completely standing in the shady area. Generally, this is again a good test area for all algorithms of the strategy.

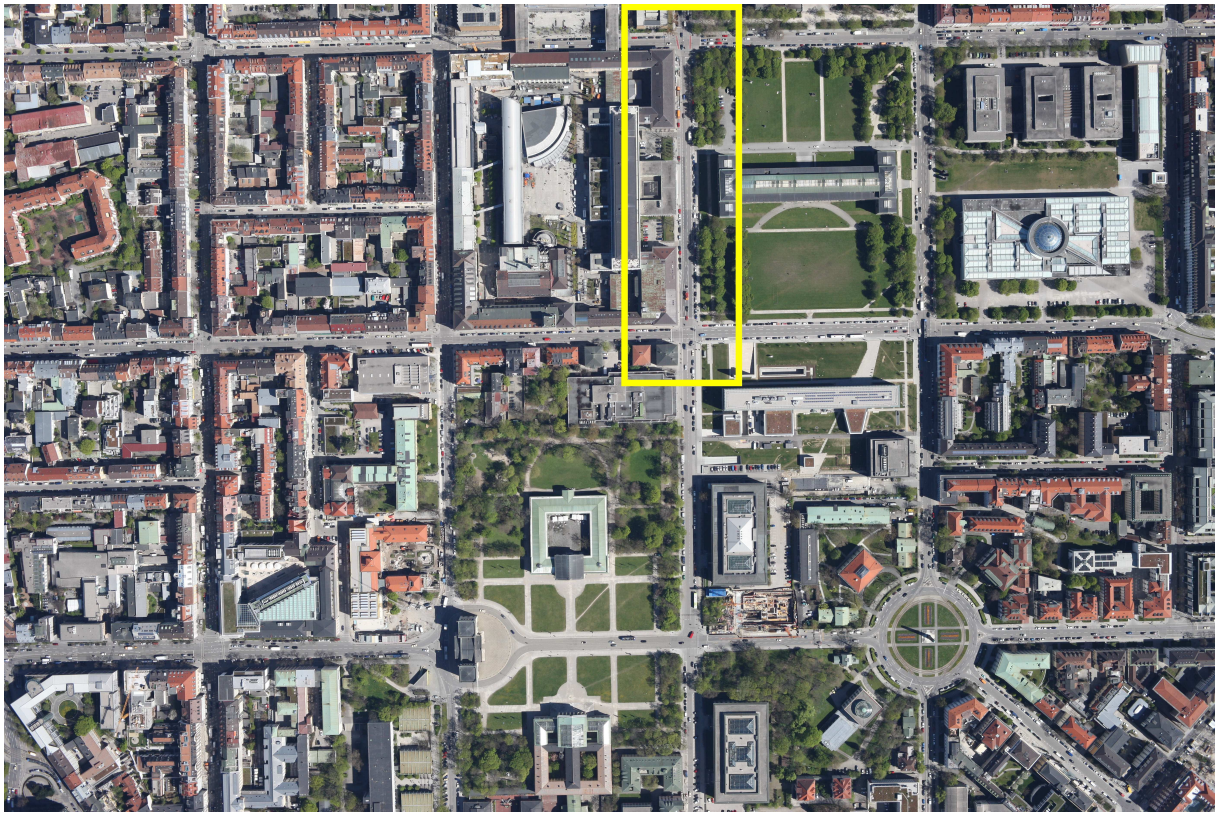


Figure 4.6: A scene of the main campus of the Technische Universitaet Muenchen (TUM) and its surrounding. The yellow rectangle indicates Dataset 4. The scene is interesting because some vehicle detection approaches have been already tested there. The scene includes heterogeneous objects with vegetation, roads and buildings; also shadow and partly occluded cars are present.

Moreover, exactly this area has been utilized to detect vehicles by a number of researchers in the past [Yao, 2010; Leitloff, 2011]. Although, sometimes the property of the base data is different (e.g., different sensor), the dataset can be perfectly used for comparing the effectiveness of the presented approach to others.

4.2.5 Dataset 5 - UltraCam, TUM, Arcisstrasse, Munich

Dataset 5 shows the same area as Dataset 4 but collected by another sensor at a different date and time. The utilized sensor was the UltraCam Eagle which usually has a better effective GSD compared to the 3K+ camera system (Section 4.1.2). However, the flying altitude at the collection time has been higher which results in an effective GSD of approximately 20 cm. Also edges are not sharp as delivered by the 3K+ system. Logically, the different time of recording leads to a different car constellations than in Dataset 4. Cars are at different positions and also the overall number of cars in the scene is lower.

Nevertheless, the scene is suitable to make the evaluation of the presented strategy more independent of the utilized sensor. Besides, it shows if the strategy is still applicable in case of reduced resolution.

4.3 Conducting the experiments

The section on conducting the experiments provides a detailed description of the conditions on which the results shown in Chapter 5 are based. This includes the determination of the optimal parameter settings and the interfaces between the methodological sections. Moreover, in some sections the expected results of the experiment are mentioned. All experiments, with the exception of Section 4.3.1, are based on the datasets introduced in Section 4.2.

4.3.1 Testing of each step considered independently

The following subsections present the conditions of the experiments for each single step of the process strategy (Chapter 3). The parameter settings are determined not regarding dependencies of preceding or subsequent steps of the strategy.

Accuracy of extracted coarse road segments

Generally, accuracies of available road databases are of broad interest for many applications, not only for vehicle extraction. Therefore, testing the accuracy of commercial and non-commercial road databases has already been carried out by several institutions (see below). Thus, in this work, road database accuracy has not been extensively tested but examples are presented where typical problems occur. In addition, the consequences for vehicle detection are discussed. Besides, I am aware of the inaccuracy of road databases, and thus the whole car extraction strategy has been designed to cope with these inaccuracies and is labeled **coarse** extraction of road segments (Section 3.1).

Furthermore, as mentioned above, several researchers published articles related to accuracy assessments of road databases. For instance, studies have been carried out to measure the positional accuracy of the OpenStreetMap (OSM) vector data combined with an enhancement solution using aerial imagery [Canavosio-Zuzelski et al., 2013; Canavosio-Zuzelski, 2013]. Additionally, another approach also based on remotely sensed imagery presents an automatic quality assessment of road database data [Gerke & Heipke, 2008].

In contrast, the following approaches are not necessarily linked to remotely sensed data. There is, for example, a comparison of the OSM and the Navteq data from Germany [Ludwig et al., 2011]. Also from regions in Germany a comparison of the OSM and the TeleAtlas data is made by Zielstra & Zipf [2010]. Whereas, Haklay [2010] compares the OSM and the Ordnance Survey dataset from England, particularly the London area. The English Ordnance survey dataset is similar to the German ATKIS.

Selection of ground regions

The experiments in this section are designed to make an assumption of how helpful disparity maps are for car detection. Especially in urban areas, cars are sometimes close to

Table 4.4: Utilized parameters for extracting the ground regions. The weighting parameters P1 and P2 are utilized in Equation 3.3. The three height parameters are used to limit the range in which the corresponding disparity is searched. The unit in meter can be simply transformed into a value in pixels using the GSD parameter.

Parameter	Parameter value
P1	750
P2	1450
GSD	13 cm
mean height	540 m
lower height	450 m
upper height	650 m

buildings. Hence, an important factor is the accuracy at the boarders from ground areas to non ground areas. In highly inaccurate cases many cars would be lost after that step.

In contrast, when to many wholes are included in the disparity map because there were no corresponding pixels found, the desired ground regions would be mixed with regions from a higher level like buildings or vegetation. This effect would lead to a higher number of false positives.

Finally, the ground regions are extracted from the disparity images. The parameter setting of the utilized SGM algorithm is shown in Table 4.4. The parameters have been chosen empirically. More information and a detailed evaluation about parameter setting for disparity map calculation can be found in Zhu et al. [2011].

Segmentation and extraction of candidate regions

A fundamental aim at the beginning of the initial development of the algorithm was to make it as general and as simple as possible. Therefore, all candidate regions, presented in Section 5.1, are processed with the same parameter setting, although some datasets are obviously different. Of course, if the parameters had been adjusted for each single dataset the outcome would have been enhanced but then the constraint of generalization would have been violated as well.

The parameter values have been empirically determined because an adequate evaluation could only be done by a human operator. The utilized values of the parameters can be found in Table 4.5.

Training of vehicle gradient classifier

The training of the classifier is carried out with 50 vertical oriented cars acting as positive samples and 2000 negative samples showing areas without cars (e.g., vegetation, roads, road markings, buildings). As previously described the size of the samples is 44×44. All

Table 4.5: Utilized parameters for extracting the candidate regions. Please refer to Section 3.3 for the corresponding equations and the detailed explanation of each parameter.

Algorithm	Parameter value(s)
Gauss filter	$\sigma = 5$
mean curvature flow	$\sigma = 1$, $\theta = 0.5$, $i = 10$
regiongrowing	$U =$ max 3
select area I	$A_1 =$ max 550
select anisometry	$An =$ max 6
select compactness	$C =$ max 4
select area II	$A_2 =$ min 100, max 150000

Table 4.6: Number of features utilized for each cascade of the classifier. The position of the specific single features is drawn in Figure 4.7

cascade no.	1^{st}	2^{nd}	3^{rd}	4^{th}
number of features	3	4	5	6

other parameters of the gradient classifier and its utilized features are explained in Section 3.4.

The training samples are taken from a different flight other than the Datasets 1–4. However, they are still from the 3K+ camera system with 13 cm GSD but of course with a slightly different illumination which is normal for a different flight. The major goal of this experiment is to show that the manual interaction can be kept very low by extracting only 50 positive training samples. The negative samples are randomly selected from a large patch of a scene which also requires low manual effort. Furthermore, no online training has been performed for which false positive detections are ported back to the training set.

Due to the low number of positive training samples and the aimed high generalization, the classifier consists only of 4 cascades and 18 features. The number of features in each cascade is shown in Table 4.6. The position of the features in the 44×44 window from the example classifier used for vertical detections in Chapter 5.1.4 and 5.2 can be seen in Figure 4.7.

The same training data has also been used for Dataset 5 from the UltraCam camera. Although, the GSD of the UltraCam data is approximately 35 % worse compared to the 3K+ camera data (20 cm → 13 cm).

The intentionally accepted drawback of the reduced training set and of the high generalization is that a satisfying result is unachievable without appropriate pre-processing steps. Hence, for reasons of comparison also results of the classifier without combining preceding steps are prepared (Section 5.1.4).

In case of Dataset 4 and 5 another additional detector has been trained and applied with an offset of 90 degrees . The reason is that cars are also oriented in the horizontal direction

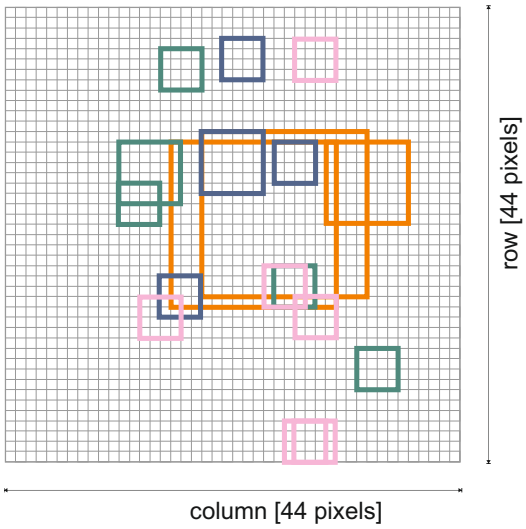


Figure 4.7: Position of each single HOG feature utilized in the example classifier. Orange, blue, green, pink rectangles are from the 1st, 2nd, 3rd, 4th cascade, respectively.

due to crossing roads. Normally, when the technique should be operationally applied, the crossing roads would be treated separately which is possible due to the integrated road databases. However, in the experiments these horizontal cars should be also detected in order to show this ability. Besides, also cars parked in the courtyard of Dataset 4 and 5 should be detected.

Vehicle classification using gradients

The classifier for vertical oriented cars is applied to Dataset 1–5 using the sliding window technique for the whole image except an certain area at the boarders of 22 pixels. In addition, in Dataset 4 and 5 also horizontal oriented cars are detected by a classifier trained with horizontal oriented cars. Finally, the classification returns a confidence image which shows the car classifier response.

Weighted selection of vehicles

The probabilities of positive detections are passed further to the weighted selection algorithm in order to eliminate multi-detections and to use the information of neighboring detections (Section 3.6.1). The region where confidence values are added is 15×15 pixels. After this step a threshold is applied to all agglomerated confidence values. However, the values are not longer normalized due to the local Gaussian weighted agglomeration.

4.3.2 Testing of complete car-detection strategy

The complete car-detection strategy is tested by carrying out the whole approach from Sections 3.2 to 3.6.1. The test Dataset 1 to 5 are utilized again. The orientation of the

roads in the datasets is according to the segments of the Navteq road database. However, the automatic use of the road segments as described in Section 3.1 has not been applied.

Unless otherwise mentioned, the used parameter setting is also the same as explained in Section 4.3.1 where each single step is tested. Please note, there is no parameter change for the Dataset 5 of the UltraCam with lower GSD. Additionally, in case of Dataset 3 no ground region estimation is used because the calculation failed as explained in Section 5.1.2.

The last two parts of the strategy (Section 3.7 and 3.8) which are marked as optional are not tested due to their very experimental state. In the case of the validation strategy (Section 3.7) the necessary fine tuning is missing. Whereas in the case of the incorporation of moving objects only a few cars are moving in the utilized datasets (Section 3.8). In addition, a method to link moving and stationary cars is still not developed. Nevertheless, these two methods are mentioned in the discussion chapter (Chapter 6).

Furthermore, the impact of the confidence threshold is shown in completeness-correctness graphs of Dataset 1, 2, 3 and 5. There is no detailed evaluation of Dataset 4 due to a missing ground truth. In this dataset many cars are at such locations which hardly allow me to count them correctly.

Finally, for the following evaluation, values like true positives (TP), false positives (FP), false negatives (FN) are used to calculate performance ratios. These ratios are correctness (Equation 4.2) and completeness (Equation 4.3). Additionally, the strictest evaluation value is the quality shown in Equation 4.4.

$$\text{correctness} = \frac{\text{TP}}{\text{TP+FP}} \tag{4.2}$$

$$\text{completeness} = \frac{\text{TP}}{\text{TP+FN}} \tag{4.3}$$

$$\text{quality} = \frac{\text{TP}}{\text{TP+FP+FN}} \tag{4.4}$$

5 Results

This chapter details the results of the experiments which are described in Section 4.3. Please note that the utilized parameter settings are also included in the previous Section 4.3. Almost all results are based on the five datasets which are introduced in Section 4.2 with the exception of Section 5.1.1. Moreover, this section shows the neutral results without evaluation – the detailed evaluation plus a discussion is then given in Chapter 6.

As mentioned, the results from every step viewed independently are first shown and then, in the next section, the final results of all steps working together are presented in combination.

5.1 Results of each step considered independently

The following results are the outcome of each single step of the strategy. Intentionally, experiments are carried out without dependence on a previous step to present each result of a single step more comprehensibly. Furthermore, also the impact of each single step becomes clearer.

5.1.1 Accuracy of extracted coarse road segments

In this section only one example is presented. The scene from the inner city of Munich is used to show an example of road extraction using vector data from road databases. The road segments of the Navteq database are drawn in red in Figure 5.1. The image is from the 3K+ camera system.

The green dashed rectangular area shows an extreme case of inaccurate road databases. When it is aimed to extract this road the symmetric buffer around the middle axis has to be as large as indicated by the green rectangle. The consequence is, if only relying on road databases to limit the search space for vehicle detection many disturbing elements from the neighboring roof have to be tackled. Generally, the overall buffer size for all road segments must adhere to the least accurate road segment.

The problematic issue for this example is not the inaccurate geocode of the image because most of the vector segments more or less fit the center of the roads. The issue is the road database itself which was originally made for navigational purposes.

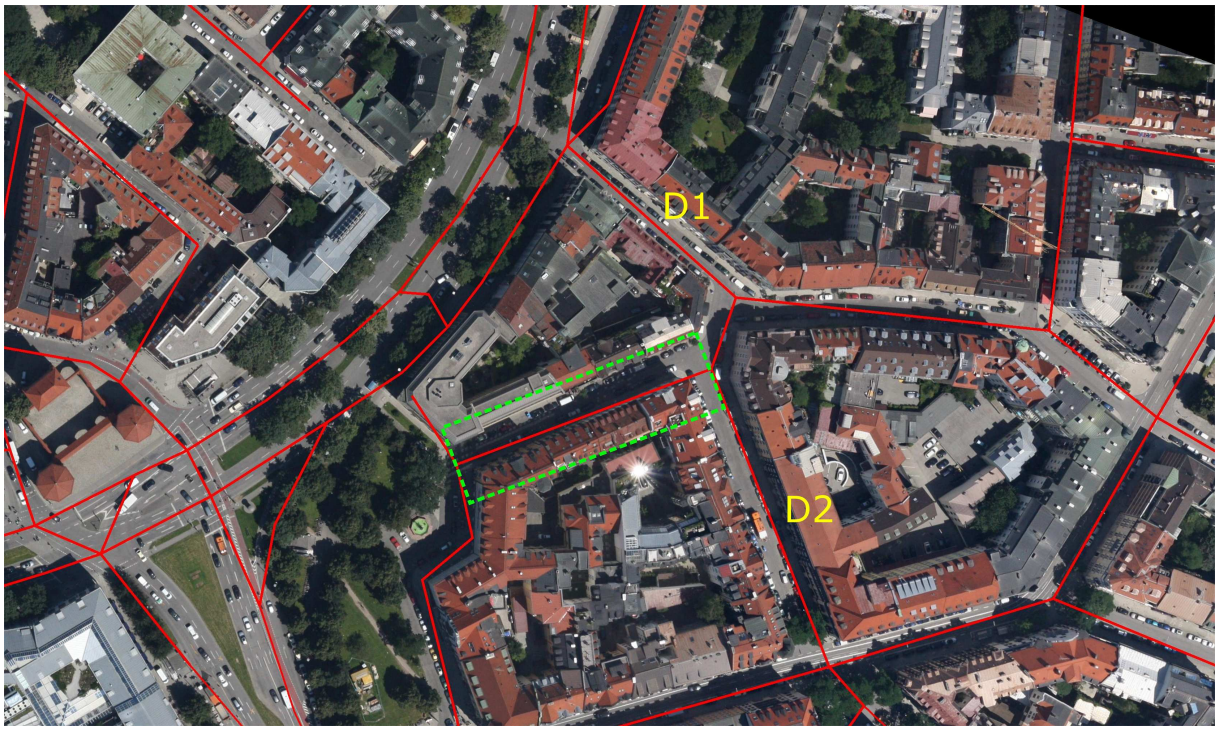
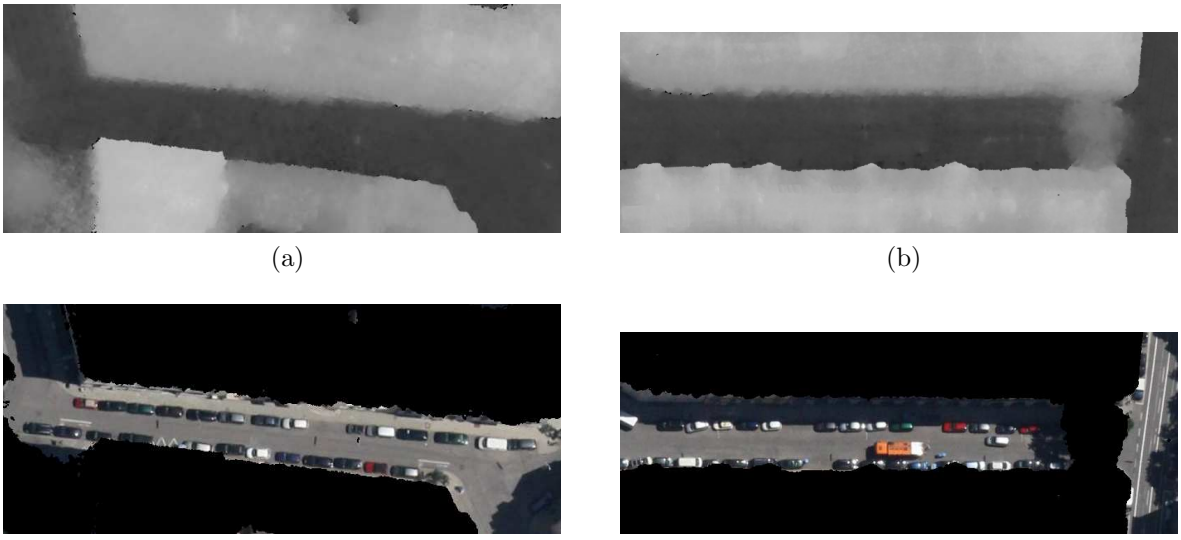


Figure 5.1: Accuracy of roads from the Navteq database in the center of Munich. The database is from the year 2008. It can be seen what problems are posed when it is aimed to only extract the road surface. The dashed green rectangle shows an extreme case of inaccuracy (All other roads must be extracted using the same buffer size as the most inaccurate one.). The normal application is routing and therefore the shown accuracy is good enough.
Additionally, the yellow D1 and D2 indicate the roads of Datasets 1 and 2, respectively.



(a) (b)

(c) (d)

Figure 5.2: Ground regions of Datasets 1 and 2. The disparity maps of Datasets 1 and 2 are shown in (a) and (b), respectively. The corresponding masked test images are then shown in (c) and (d). The ground regions are determined by applying the previously calculated threshold to the disparity maps. Finally, those masks are overlaid onto the original images.

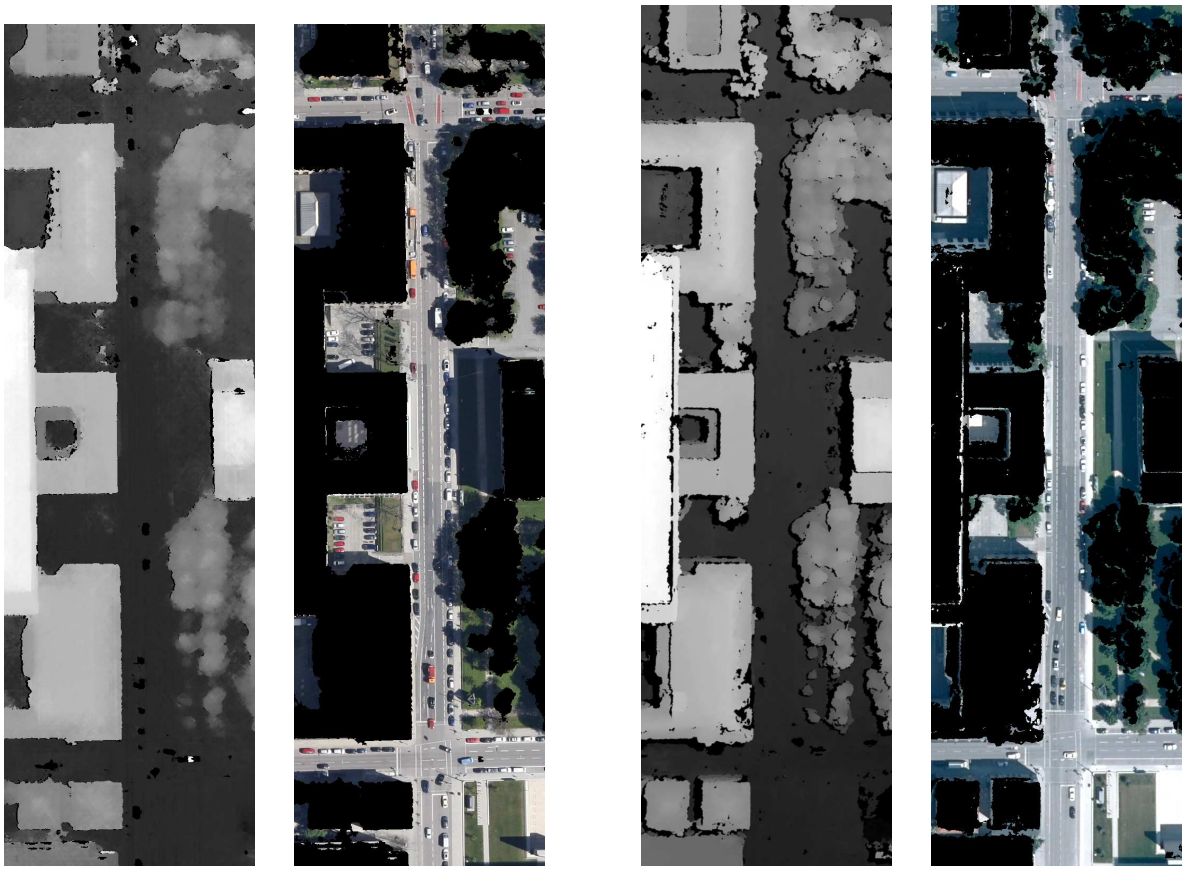
5.1.2 Selection of ground regions

The results of the algorithm to extract ground regions (Section 3.2) are shown here. In Figure 5.2a and Figure 5.2b the calculated disparity images for Datasets 1 and 2 are shown, respectively. In addition, in Figure 5.2c and Figure 5.2d, the original search images are overlaid by the calculated masks to show which parts of the images remain and what accuracy can be obtained.

The same algorithm applied to Dataset 4 and 5 leads to the results presented in Figure 5.3. Please note that the baseline of the two consecutive images used for calculating the disparity map of Dataset 5 is longer compared to Datasets 1 to 4 due to a lower recording frequency.

Finally, the four different graphs received from the Minimum Error Thresholding for Datasets 1, 2, 4 and 5 are shown in Figure 5.4. The minimum of each graph is the threshold which decides whether the area is at ground level or not. The effective values are displayed in Table 5.1. In the same Table the ratio of the remaining ground area to the original area is also included.

The result of Dataset 3 is treated separately because the algorithm failed due to areas below ground level (Figure 5.5b). These areas have been caused by a construction site and lead to a malfunction of the Minimum Error Thresholding algorithm (Figure 5.5a). As can be seen the determination of a correct minimum is impossible.

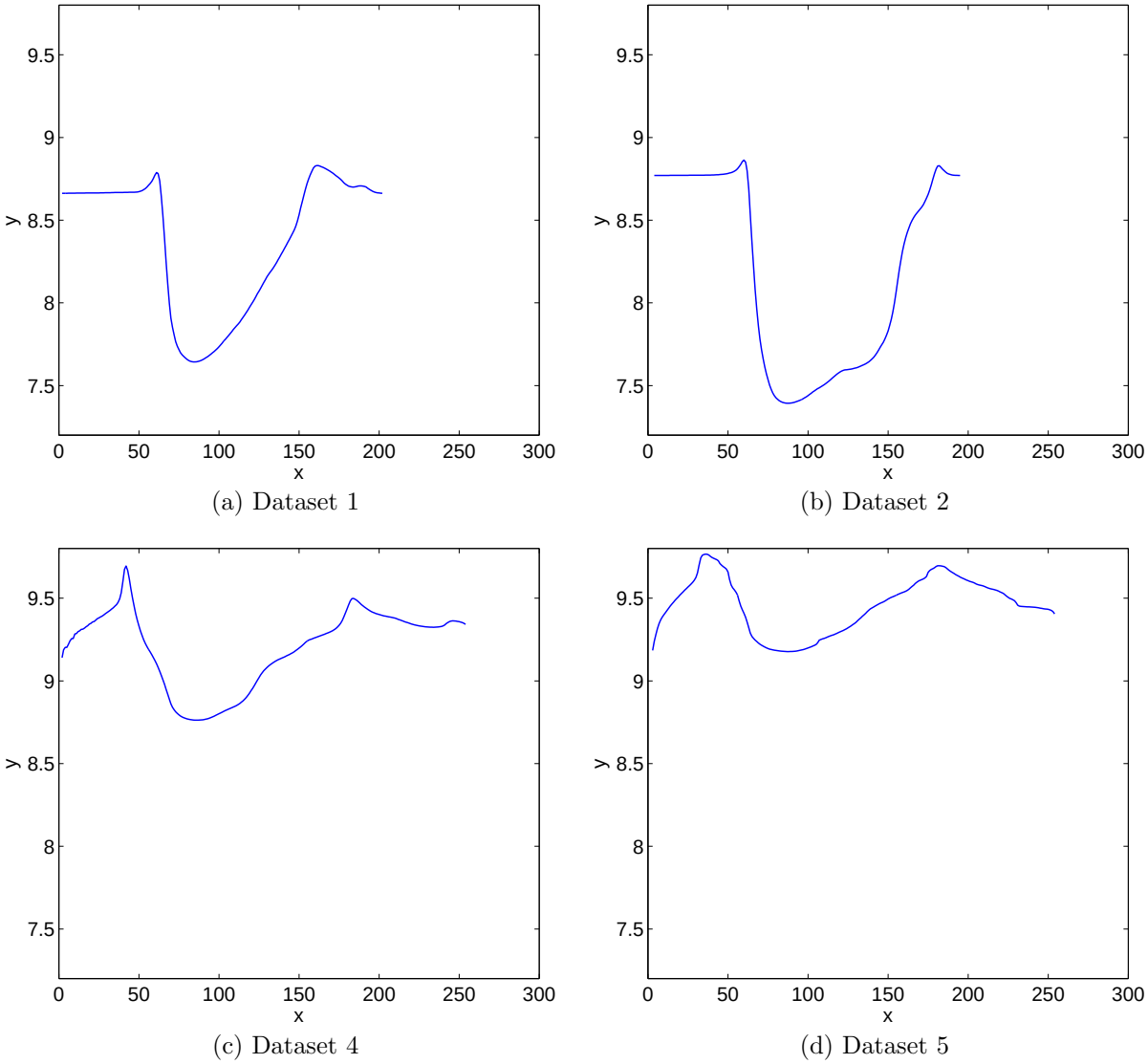


(a) (b) (c) (d)

Figure 5.3: Ground regions of Datasets 4 and 5. The disparity maps of Datasets 4 and 5 are shown in (a) and (c), respectively. The corresponding masked test images are then shown in (b) and (d). The ground regions are determined by applying the previously calculated threshold to the disparity maps. Finally, those masks are overlaid onto the original images.

Table 5.1: Statistics of ground region extraction. The minimum errors of Datasets 1 to 5, received from Equation 3.4. These errors are used as threshold to distinguish between ground and non-ground areas. Finally, it is applied to the histogram equalized disparity map. The second row shows the ratio of the ground area to the original image which could be successfully excluded from the subsequent car detection process.

	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
minimum error (threshold)	84	87	15	85	87
remaining area [%]	36	37	n/a	53	49



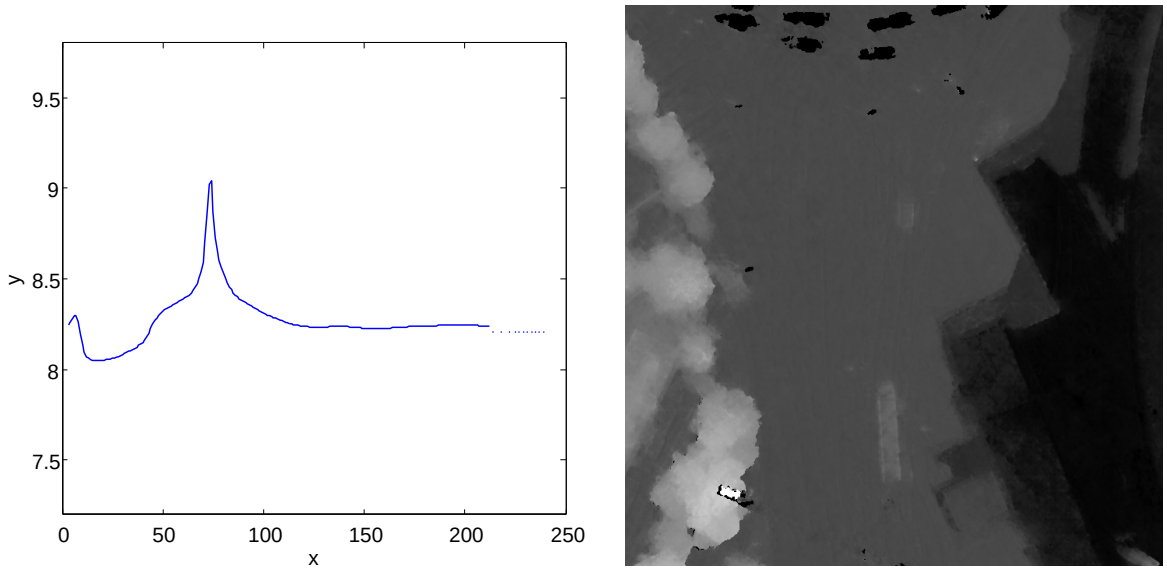
(a) Dataset 1

(b) Dataset 2

(c) Dataset 4

(d) Dataset 5

Figure 5.4: Graphs resulted from the Minimum Error Thresholding (Equation 3.4). The x-axis indicates the value of the threshold ranging from 5 to 251 (not 1 to 256 because a limit is used). The y-axis indicates the result of Equation 3.4 without *argmin* at every x position. The global minimum of the graphs is the value of the 8 bit disparity map which separates ground from non-ground area. The minimum is very clear for every utilized dataset. Minima in the range of very low values (0-5) are excluded from the process because they can cause a wrong result. Values in that range are often from areas in the disparity image where no match was found.



(a) (b)

Figure 5.5: Graph resulted from the Minimum Error Thresholding – Dataset 3. (a) Result of Equation 3.4. (b) Disparity map of Dataset 3. The construction site included in Dataset 3 leads to a result which shows limitations of the particular ground selection strategy. The area of the construction site is below road level, thus the Minimum Error Thesholding sets the Minimum to 15 which is the threshold between underground level and ground level. In this case the presented strategy is not suitable and alternative ways have to be discovered. Alternative methods are, for instance, presented in Sezgin & Sankur [2004]. However, the occurrence of large regions below ground level in dense urban areas is relatively seldom.

5.1.3 Segmentation and extraction of candidate regions

The extraction of candidate regions is shown stepwise for every single dataset. The results for Datasets 1, 2, 3, 4 and 5 are shown in Figure 5.6, 5.7, 5.8, 5.9 and 5.10, respectively. In each Figure, part (a) shows the image after the mean curvature flow. It can be seen that contours of objects are preserved but the color gradient of the objects is smoothed. Subsequently, subfigure (b) shows the result of the color region-growing (Section 3.3.2) and subfigure (c) and (d) show the outcome of the select area I algorithm (Section 3.3.2) and the select area II algorithm (Section 3.3.2), respectively. The final candidate regions are then presented in (e) after the intersection step.

Each color describes one region. The different colors are for visualization reasons. They are randomly chosen and have no deeper meaning. Finally, in the optimal case, every region is an object like a single car or a line of cars.

The necessity of the last intersection step (e) can be well seen by comparing Figure 5.6d and Figure 5.6e (lower left corner). Also the upper left corner of Figure 5.7d shows a location where the intersection step was helpful. The second region growing before (d) leads to new undesired regions which have to be removed again. However, this last step was not necessary in the case of the other datasets.

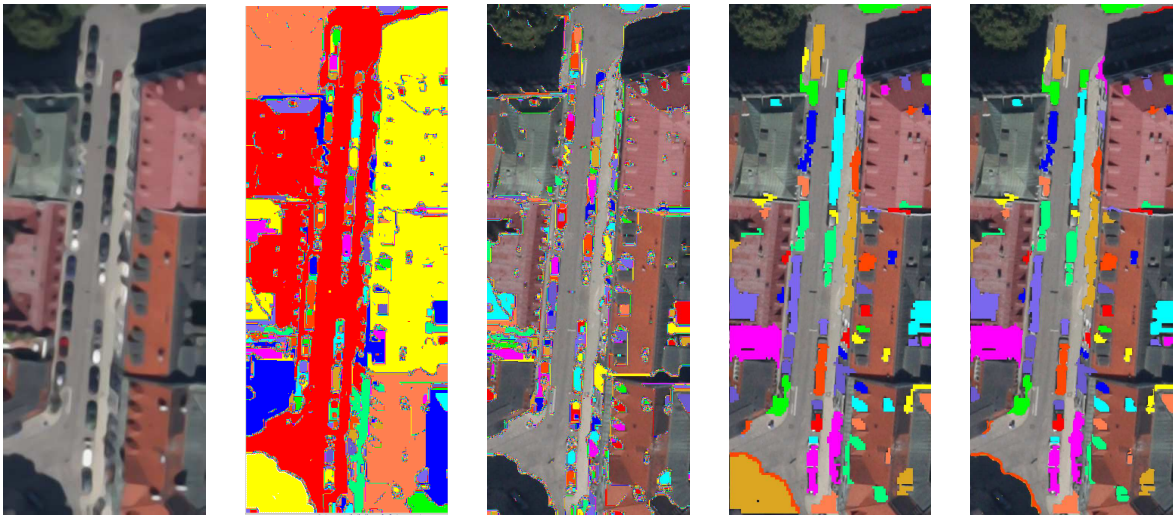
An analytical contribution is illustrated in Table 5.2. The impact of each step in total pixels and in percent relative to the original area is presented. Furthermore, a second Table 5.3 shows the quality of the extraction. The table includes numbers of cars which are lost after the candidate extraction step and under which conditions this occurred. For instance, areas with special light conditions like shadow areas or partly shadow areas are of special interest. These areas can pose problems when the usage of a fix parameter setting is desired.

5.1.4 Vehicle classification using gradients

The results of the application of the HOG feature-based classifier are presented in the following figures. Datasets 1 and 2 are classified as follows (Figure 5.11). These two datasets are presented in the same figure due to their similarity. Results of Dataset 3 can be seen in Figure 5.12. Furthermore, the outcome after the classification of Dataset 4 is shown in Figure 5.13 and the result of Dataset 5 is presented in Figure 5.14.

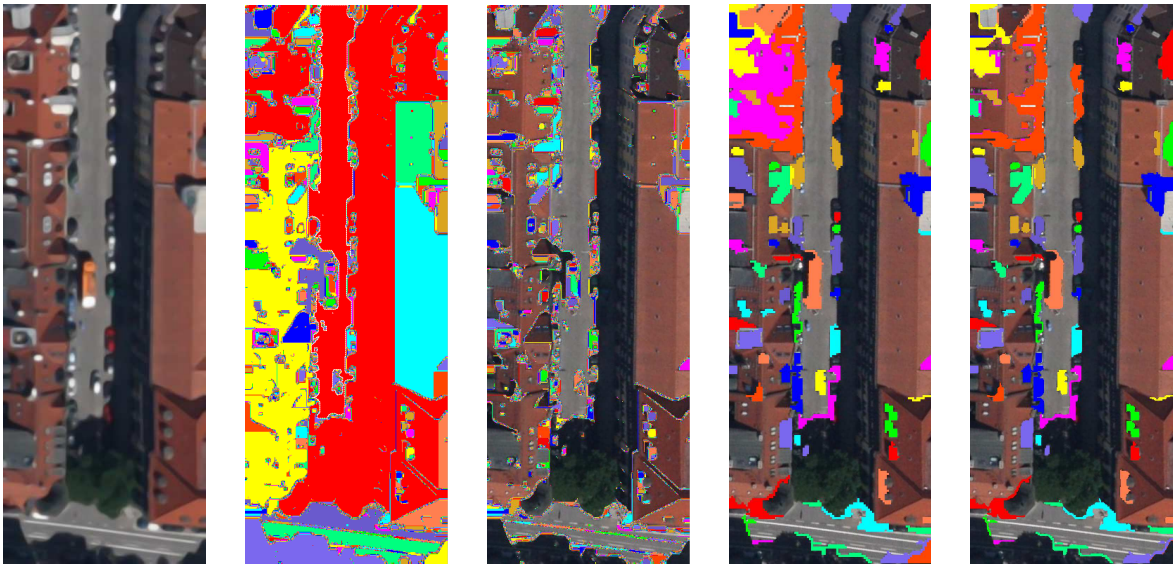
Objects which are assumed to be cars are marked with a red cross. The complete confidence map showing responses of the classifier for each pixel is always presented on the right side in the figures. The color code can be interpreted in the following way: a high positive value indicates that the classifier is very confident that there is a car at this position (the typical color range is from yellow to red). In contrast, negative values like turquoise and blue indicate that the pixel belongs to a non-car object. It is good to see which areas are wrongly identified by the classifier.

The threshold of the confidence values leading to the marked detections in Figures 5.11a, 5.11c, 5.12a, 5.13a and 5.14a is set manually, in order to get a visual impression. An automatic threshold can be determined by considering the information of Figure 5.18. A



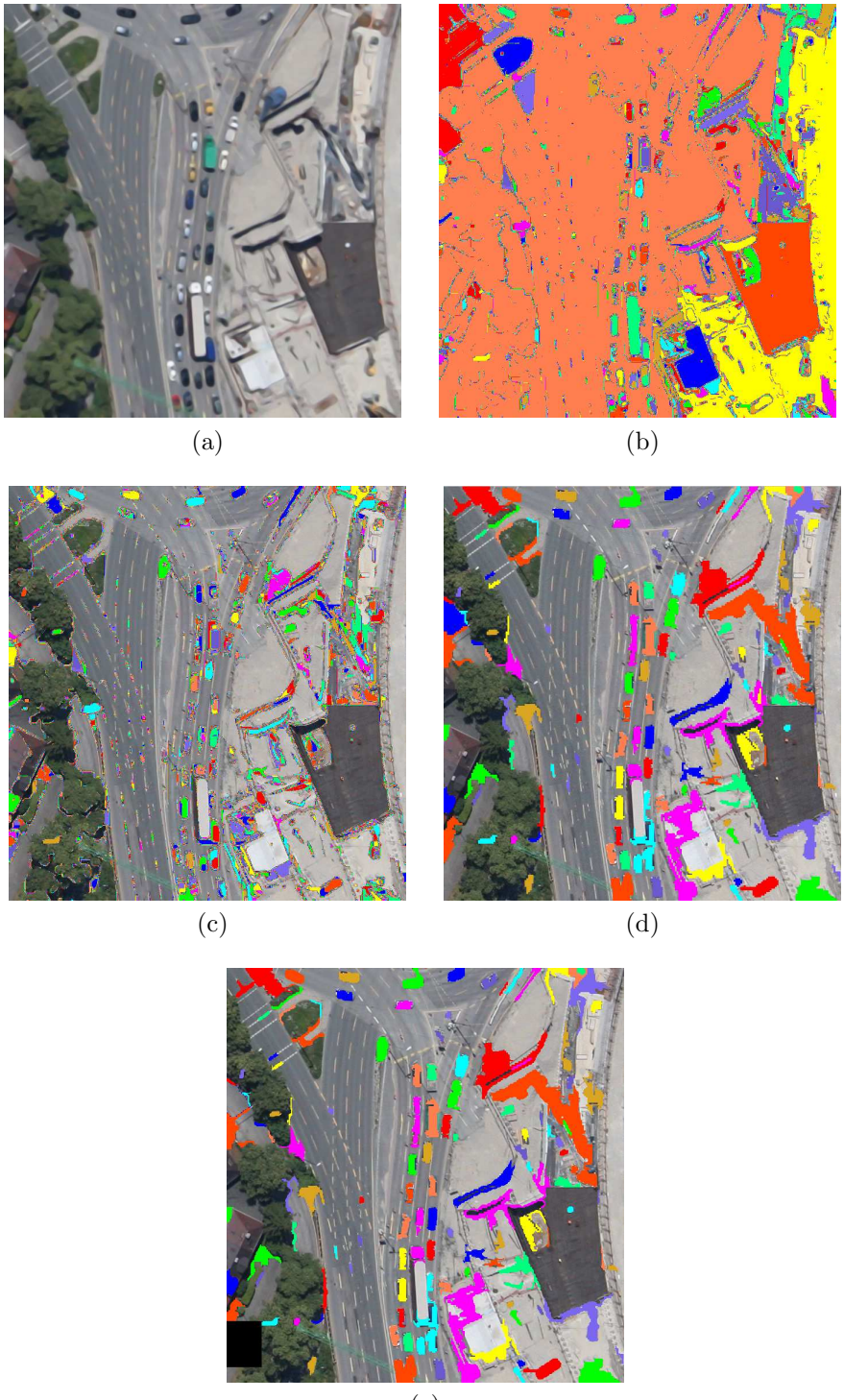
(a) (b) (c) (d) (e)

Figure 5.6: Segmentation and extraction of candidate regions applied to Dataset 1. (a) Mean curvature flow. (b) Region growing. (c) Selection area. (d) Selection area II. (e) Intersection.



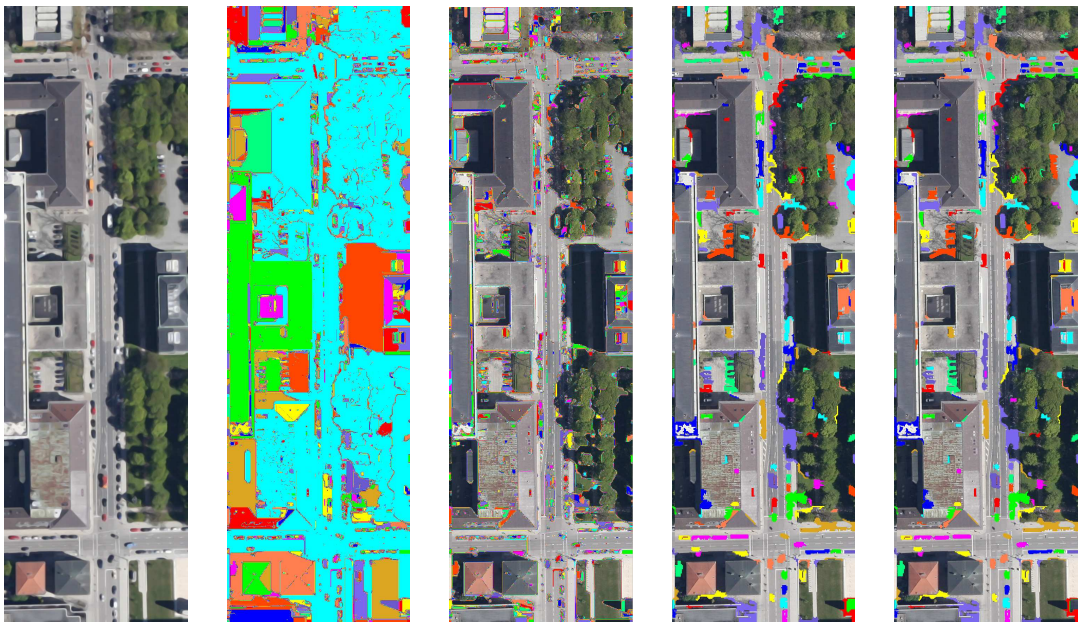
(a) (b) (c) (d) (e)

Figure 5.7: Segmentation and extraction of candidate regions applied to Dataset 2. (a) Mean curvature flow. (b) Region growing. (c) Selection area. (d) Selection area II. (e) Intersection.



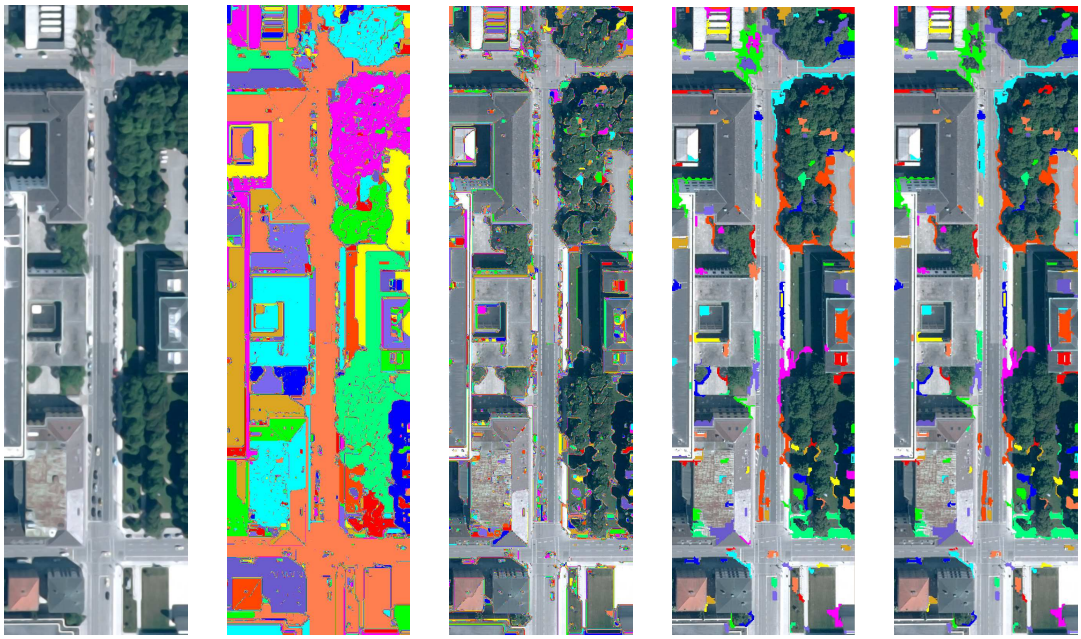
(a) (b) (c) (d) (e)

Figure 5.8: Segmentation and extraction of candidate regions applied to Dataset 3. (a) Mean curvature flow. (b) Region growing. (c) Selection area. (d) Selection area II. (e) Intersection.



(a) (b) (c) (d) (e)

Figure 5.9: Segmentation and extraction of candidate regions applied to Dataset 4. (a) Mean curvature flow. (b) Region growing. (c) Selection area. (d) Selection area II. (e) Intersection.



(a) (b) (c) (d) (e)

Figure 5.10: Segmentation and extraction of candidate regions applied to Dataset 5. (a) Mean curvature flow. (b) Region growing. (c) Selection area. (d) Selection area II. (e) Intersection.

Table 5.2: Statistics of the segmentation procedure. It is shown how many pixels are excluded after each processing step. The first number is the absolute number of pixels while the second one relatively expresses the result in percent. The description of the single algorithms can be found in Section 3.3.

Number of pixels	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
at the beginning	181436	230112	497520	911040	773568
[%]	100	100	100	100	100
after select area	42305	50760	106511	162461	152128
[%]	23	22	21	18	20
after select anisometry	41169	48924	101976	150472	140521
[%]	23	21	21	17	18
after select compactness	37331	44747	97685	127121	112072
[%]	21	19	20	14	15
after intersection	29301	38055	73165	107001	84572
[%]	16	17	15	12	11

reasonable compromise between completeness and correctness must be found which also depends on the kind of application (Section 6.1.1).

5.2 Results of complete car-detection strategy

The final results of the whole car detection strategy for Datasets 1 and 2 are illustrated in Figure 5.15. Moreover, the result for Datasets 3, 4, 5 is in Figure 5.16, 5.17a, 5.17b, respectively.

The impact of the threshold for the classifier is indicated by the completeness-correctness curves in Figure 5.18. The utilized thresholds for the visualized results are a trade-off between completeness and correctness. There is no numeric value given because it is not normalized. The reason for this is the Gaussian weighting procedure which returns not-normalized thresholds and requires a threshold which is not between 0 and 1.

Finally, the achieved quality of the overall strategy can be seen in Table 5.4. The quality value is calculated using Equation 4.4.

Table 5.3: Statistics of the segmentation procedure II. It is shown how many cars are lost due to the segmentation procedure. Additionally, lost cars in shadow areas are listed as well.

Number of cars	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
in the scene	28	37	36	136	67
[%]	100	100	100	100	100
in shadow area	3	2	0	18	7
[%]	11	5	0	13	10
partly in shadow area	0	15	0	13	11
[%]	0	41	0	10	16
lost after segmentation	3	3	0	13	3
[%]	11	8	0	10	4
lost and in shadow area	3	1	0	11	2
[%]	11	3	0	8	3
lost and partly in s. a.[a)]	0	2	0	2	1
[%]	0	5	0	0	1

[a)] s. a. = shadow area

Table 5.4: Maximum quality of the final results. The quality value is the strictest value for evaluating the results because false positive and false negative detections are included in one number. The calculation is done by using Equation 4.4.

	Dataset 1	Dataset 2	Dataset 3	Dataset 5
maximum quality	82 %	70 %	63 %	64 %

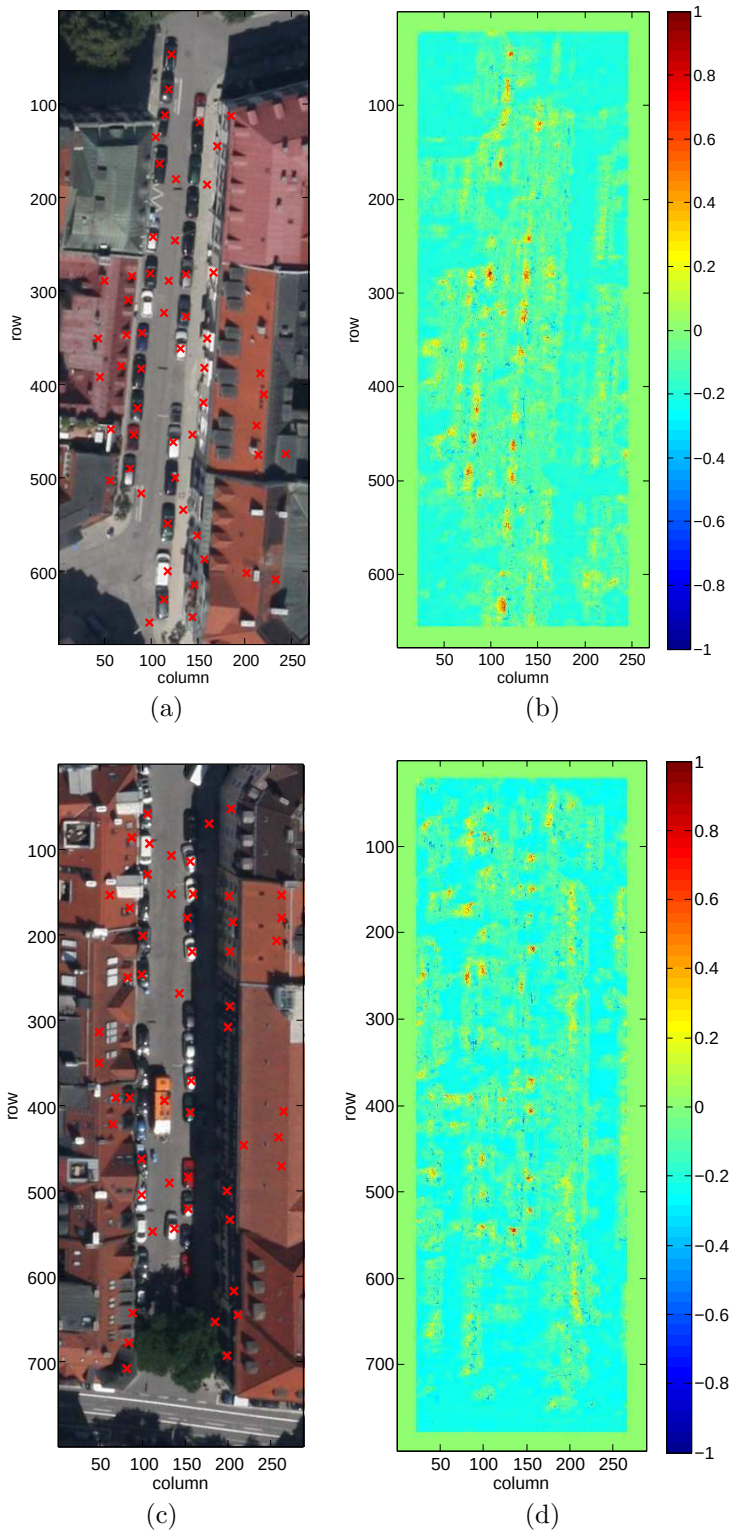


Figure 5.11: Gradient-based classification of Datasets 1 and 2. (a) Dataset 1. The objects assumed to be cars are marked with a red cross. (b) Confidence map of Dataset 1. Positive values (yellow to red) indicate a detected car while negative values (turquoise to blue) indicate a different object than a car. (c) Dataset 2. (d) Confidence map of Dataset 2.

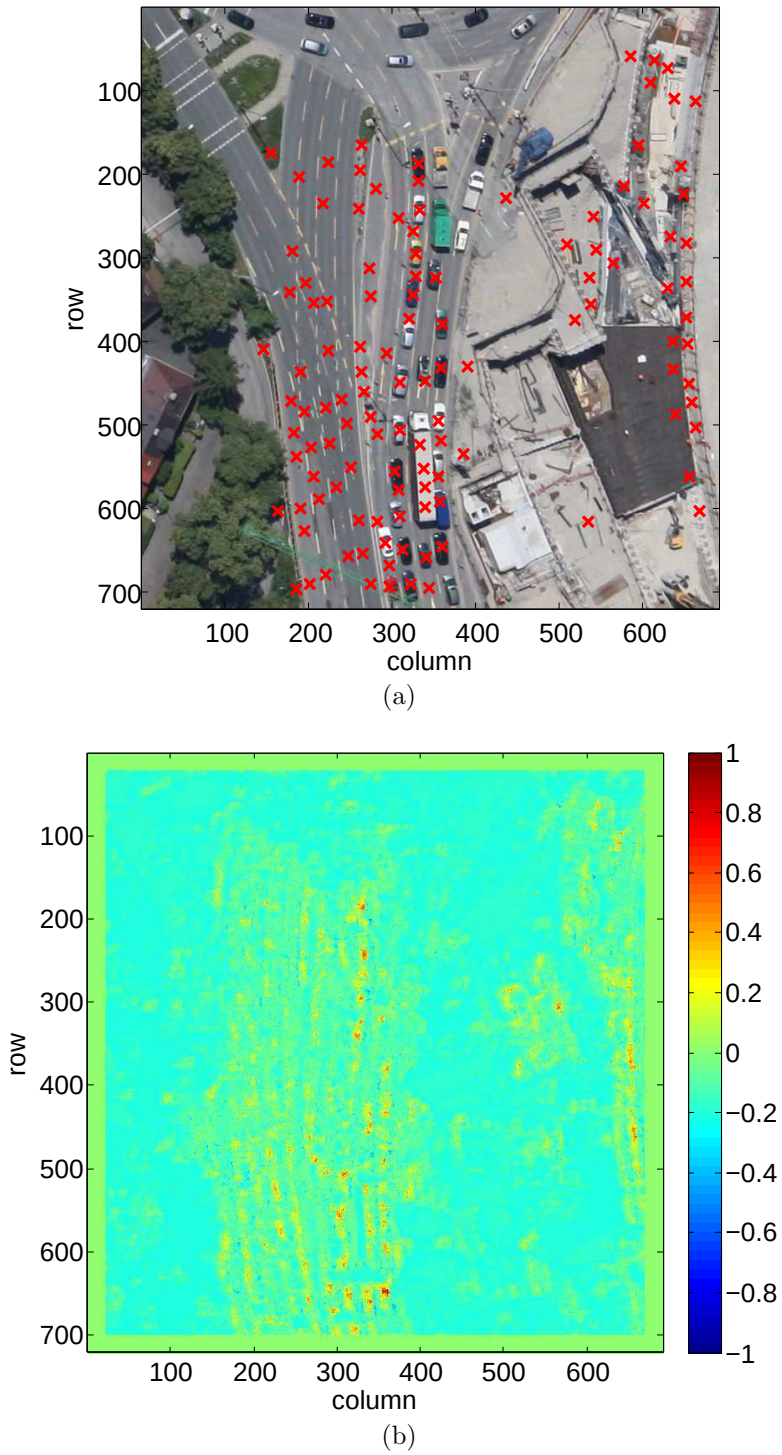


Figure 5.12: Gradient-based classification of Dataset 3. (a) Dataset 3. The objects assumed to be cars are marked with a red cross. (b) Confidence map of Dataset 3. Positive values (yellow to red) indicate a detected car while negative values (turquoise to blue) indicate a different object than a car.

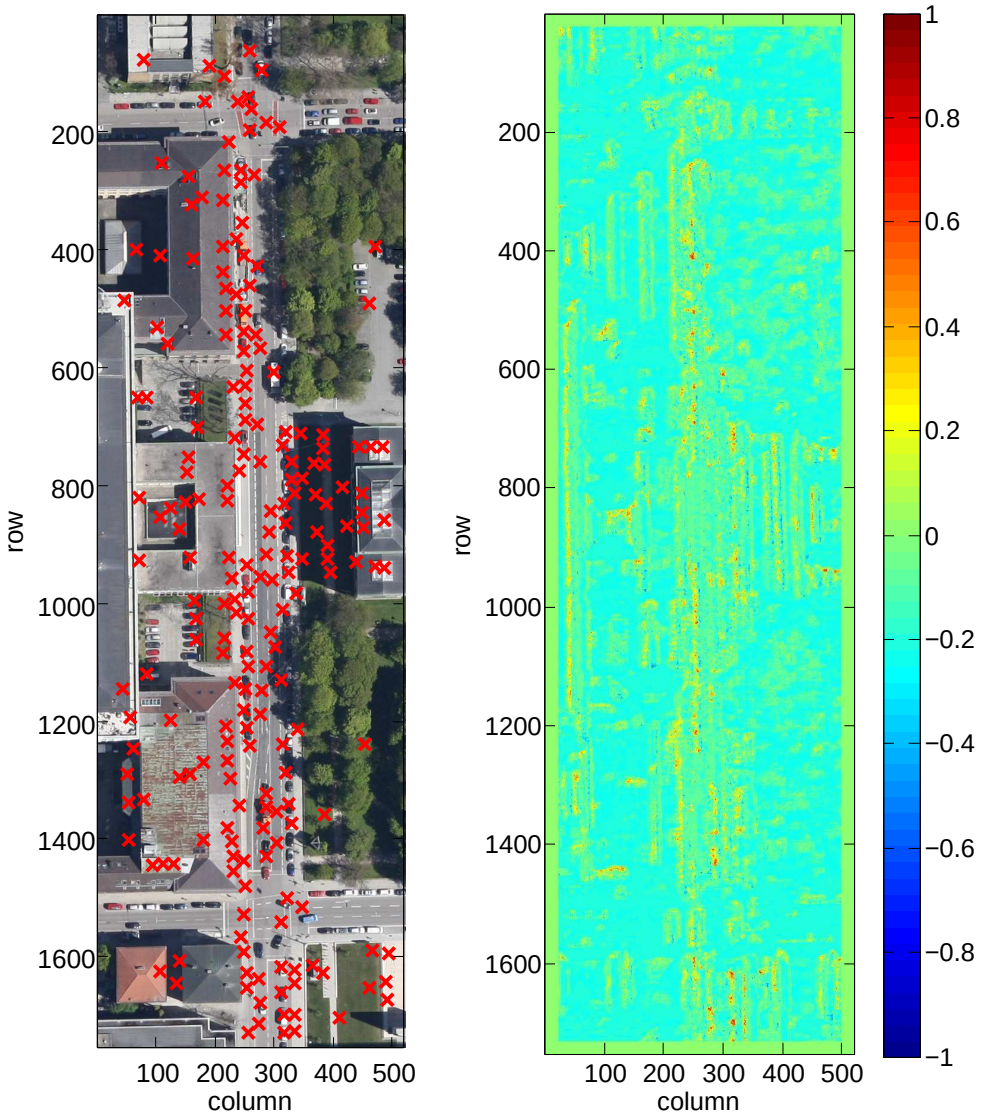


Figure 5.13: Gradient-based classification of Dataset 4. (a) Dataset 4. The objects assumed to be cars are marked with a red cross. (b) Confidence map of Dataset 4. Positive values (yellow to red) indicate a detected car while negative values (turquoise to blue) indicate a different object than a car.

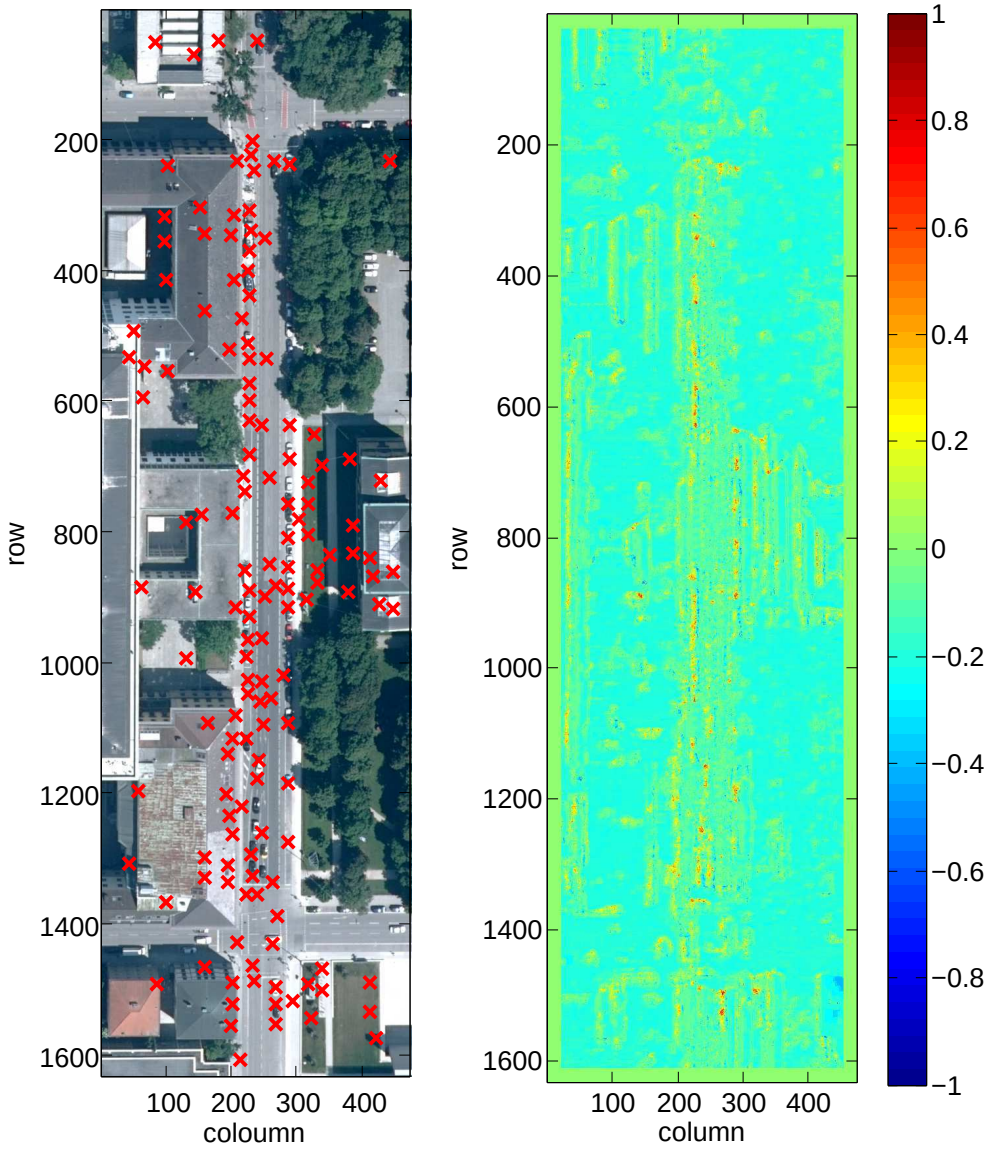
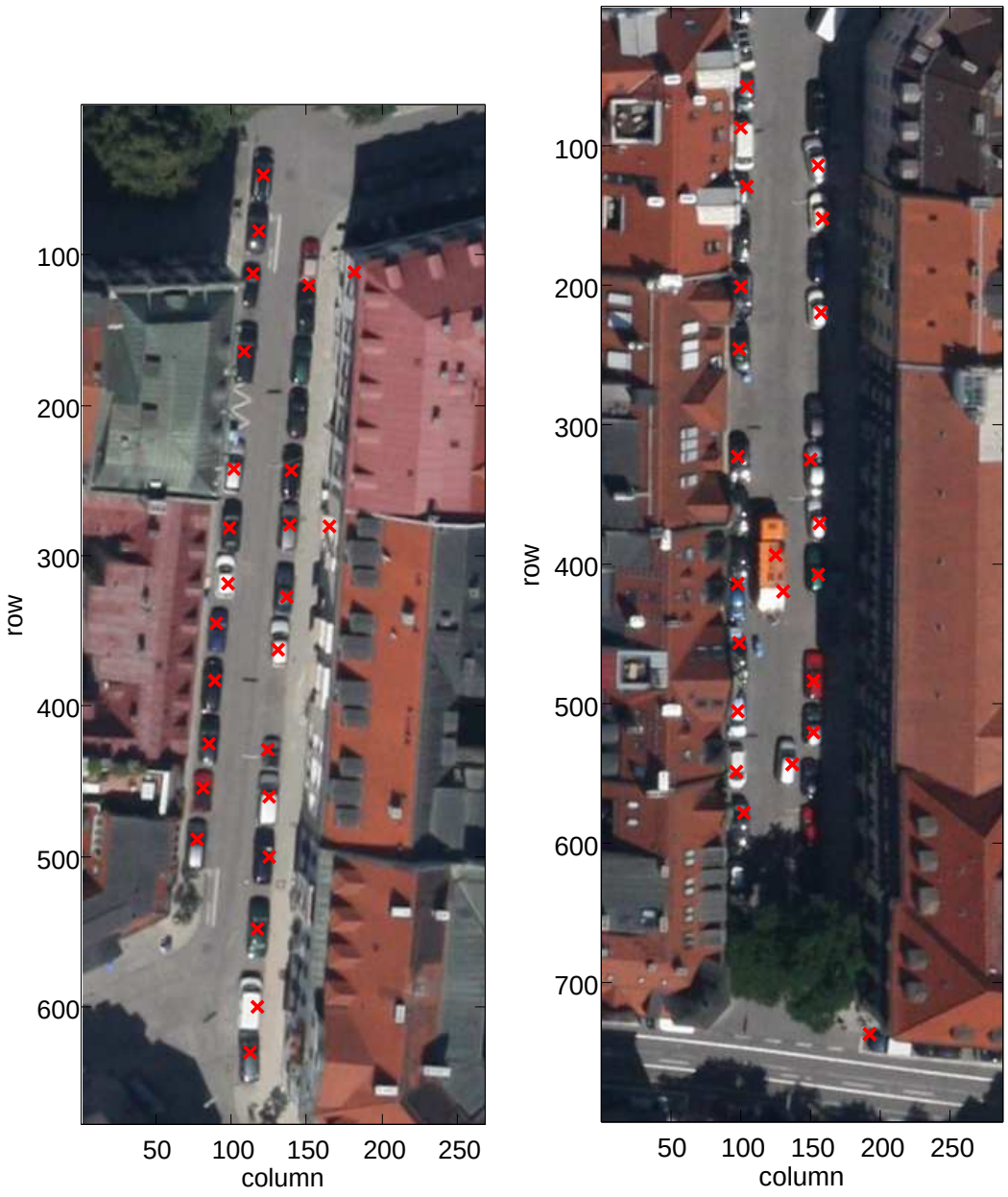


Figure 5.14: Gradient-based classification of Dataset 5. (a) Dataset 5. The objects assumed to be cars are marked with a red cross. (b) Confidence map of Dataset 5. Positive values (yellow to red) indicate a detected car while negative values (turquoise to blue) indicate a different object than a car.



(a) Dataset 1 (b) Dataset 2

Figure 5.15: Final result of Dataset 1 and 2. Red crosses indicate a vehicle candidate.

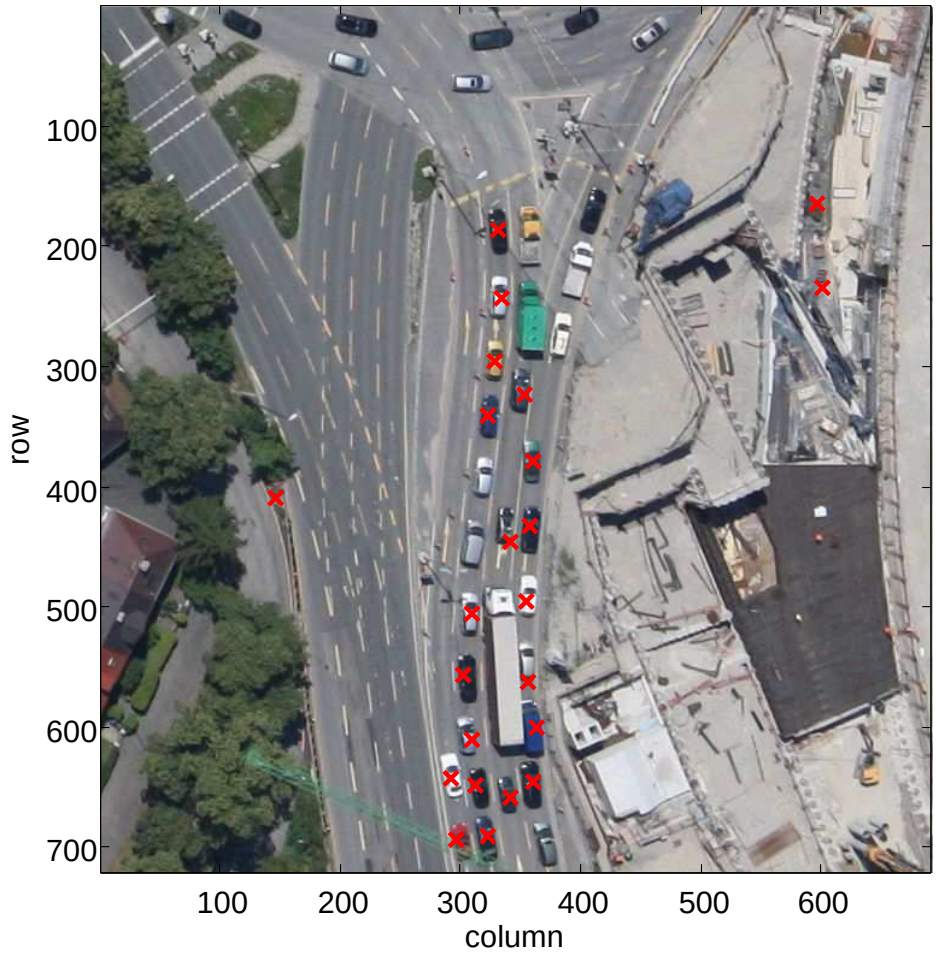
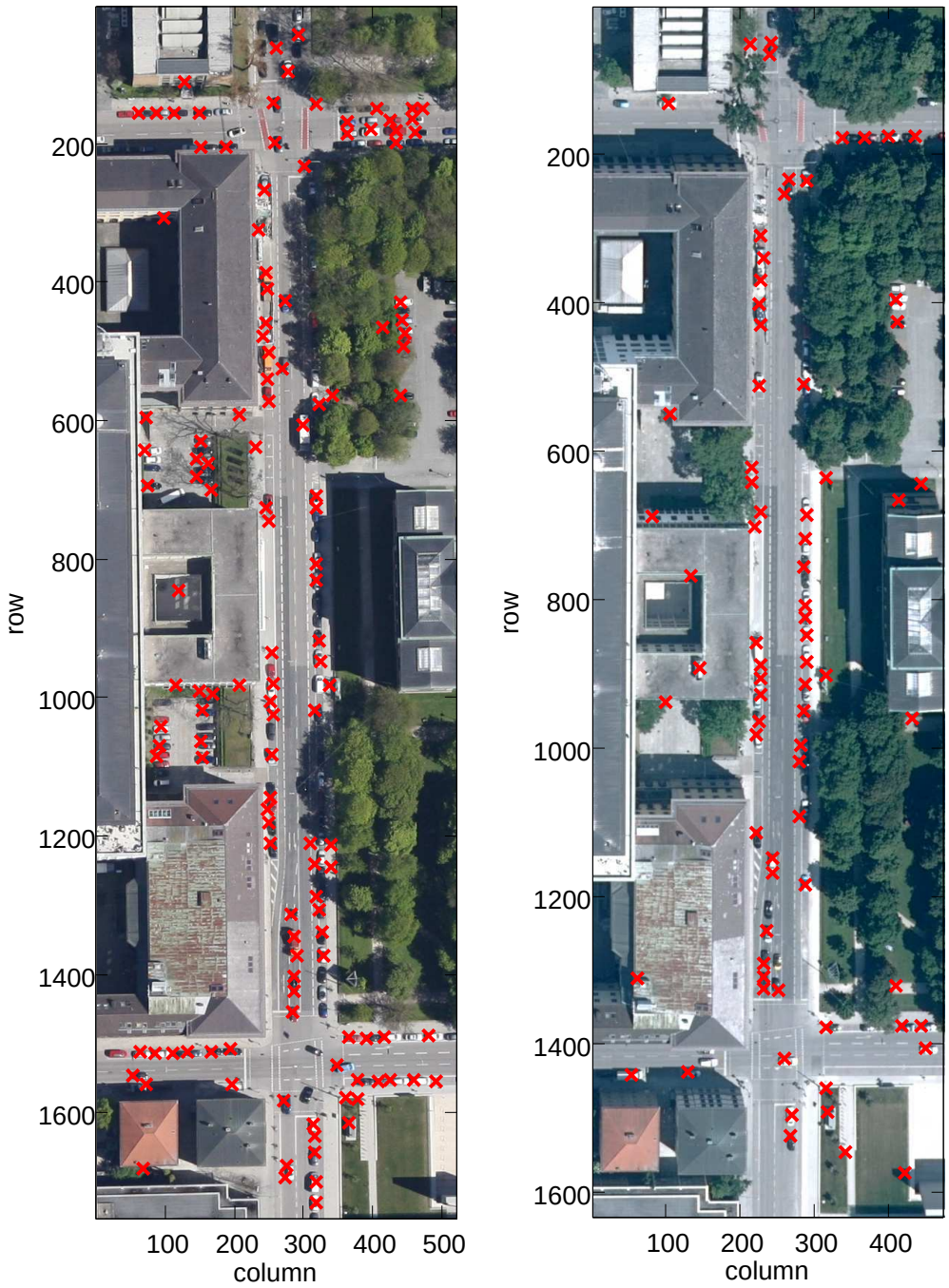


Figure 5.16: Final result of Dataset 3. Red crosses indicate a vehicle candidate.



(a) Dataset 4

(b) Dataset 5.

Figure 5.17: Final result of Datasets 4 and 5. Red crosses indicate a vehicle candidate.

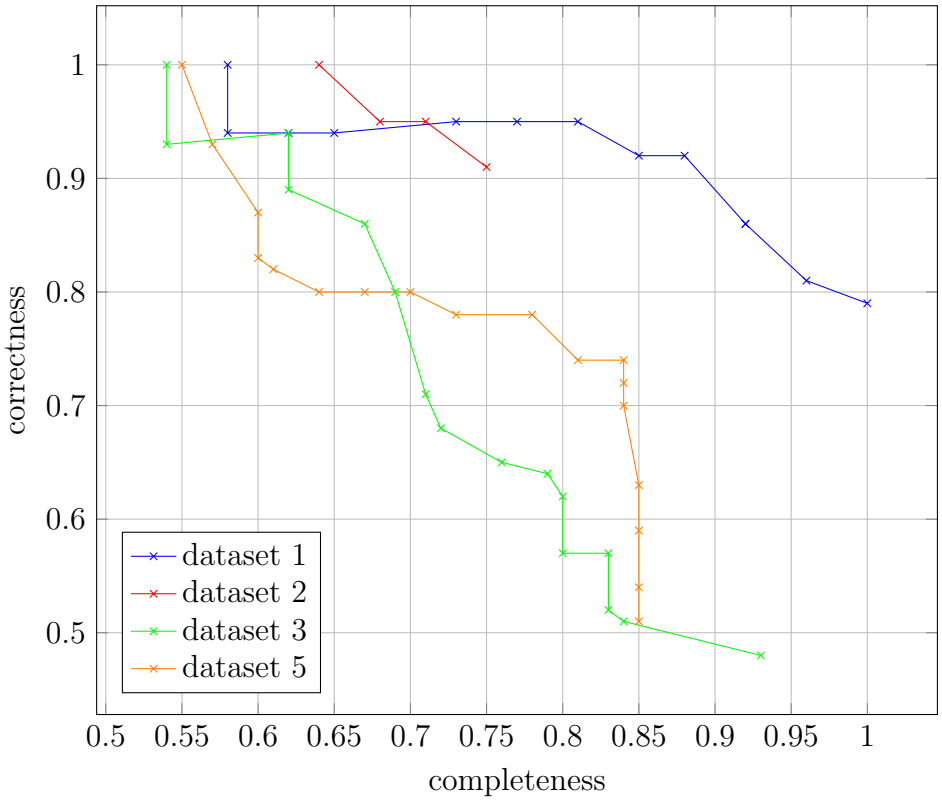


Figure 5.18: The graphs give an impression how the algorithm performs when the final threshold of the classifier is adjusted. The completeness is plotted on the x-axis and the correctness is plotted on the y-axis. The definition of correctness and completeness can be found in Equation 4.2 and 4.3.

6 Discussion

The results in Chapter 5 are discussed in the following sections. Moreover, the organization is also done according to Chapter 5. Thus, results of every step are discussed individually, followed by a commentary on the complete strategy.

6.1 Discussion of each step considered independently

The single steps are structured in the same way as in Section 5.1. In addition, comments on the optional methods (Section 3.7 and 3.8) are also included.

6.1.1 Accuracy of extracted road segments

As mentioned before, the accuracy of road databases is often not good enough to mask out a road precisely (Figure 5.1), and thus many objects in the surrounding road area have usually to be accepted in the detection process (Section 5.1.1).

In this work road databases have been only applied to get a coarse target area and to speed up the calculation time of the disparity image. However, detecting cars in densely populated cities without road databases is not a problem as shown in the next section. Moreover, the situation changes when traffic from roads which are not surrounded by buildings is monitored (e.g., highways). Then the use of road databases could be very helpful.

6.1.2 Selection of ground regions

The simultaneous calculation of disparity images turned out to be a very good supporting strategy for car detection. Especially, a very reliable ground area could be extracted from Dataset 1 and 2 (Figure 5.2c and 5.2d). The separation between ground and non-ground levels is very precise which enables the subsequent detection algorithm to detect all cars close to the roof (Figure 5.15a and 5.15b). Additionally, problematic areas like façades with rectangular windows are also partly extracted. Rectangular windows are often classified as false positives due to their similar contour like cars (see Figure 5.11a and 5.11c).

Unfortunately, the experiment failed in case of Dataset 3 (Figure 5.5a). The reason is that a combination of a large flat area on ground level and an area below ground level poses problems to the thresholding procedure (Section 3.2.2). The algorithm assumes that the

lowest area of the patch is the desired ground area. Conclusively, the simple strategy using the minimum error thresholding is not suitable for such advanced problems. A solution is required which covers all possible situations and checks if the returned ground area is the actual ground level.

Furthermore, the resulting disparity maps of Dataset 4 and 5 (Figure 5.3b and 5.3d) are helpful for the subsequent car extraction step but the result is not optimal. This can be explained by the fact that the sample scene is not optimal for the objective of the proposed method due to large flat regions non-trafficable (e.g., the lawn in the lower right corner of Figure 5.3b and 5.3d). These regions pose problems to the detector because than also objects from non-trafficable areas have to be considered in the subsequent car extraction approach which are not used in the training procedure. Thus, more potential false positives are included. Highly densely populated inner city areas like Paris, France are more optimal. Moreover, a difference between the disparity maps from Dataset 4 and 5 (Figure 5.3a and 5.3c) regarding the quality can be observed. This mainly stems from the longer baseline of Dataset 5 (distance from sensor position one to sensor position two) resulting from the lower imaging frequency of the UltraCam compared to the 3K+ camera system (Table 4.2). A longer baseline leads to not all objects being included in both images – especially as far as façades or general vertical planes are concerned. The consequence is that more holes are in the disparity map of Dataset 5. In order not to lose any potential cars, these holes are always classified as ground area which finally leads to a less accurate ground level mask.

Generally, an important fact, valid for all datasets, is that no car got lost due to the application of disparity maps (see Figure 5.2c, 5.2d, 5.3b and 5.3d). The required calculation time might be the only drawback. For instance, the calculation of a disparity map can be seen as a single instruction multiple data (SIMD) problem and hence an alternative implementation using the Graphics Processing Unit (GPU) is many times faster.

Moreover, moving cars are in most cases not a problem because the distance between their position in the first and in the second image is often so great that the disparity is not calculated at this position. The maximum disparity is a parameter which corresponds to the upper height and lower height as shown in Table 4.4. Finally, holes at these locations are treated as ground area and thus do not disturb the subsequent procedure. However, problems could occur when a car has a certain velocity which leads to a reasonable disparity and a high object such as a tree or a house is assumed. Consequently, the location is then masked out but this situation was very rarely observed.

An additional strategy is to exploit the height of the cars in the disparity image. Approaches which follow that idea are well known in the field of car extraction from LiDAR data [Toth, 2009; Yao et al., 2011]. The ability to exploit the height of the cars in the test datasets with a resolution of 13 cm and 20 cm has to be proven.

To this end, the substitution of road databases by disparity images is a very good alternative in dense urban areas (Dataset 1 and 2, Figure 5.2) where up to 64 % of the original image could be excluded. It is more difficult to leave out road databases in flat areas (Dataset 3, Figure 5.5b) and in partly flat areas (Dataset 4 and 5, Figure 5.3) where up to 47 % could be excluded. Important to note is that the application of disparity maps never led to an exclusion of a car.

6.1.3 Segmentation and extraction of candidate regions

The novel approach to extract car candidates based on mean curvature flow and region growing can be evaluated as follows.

Generally, the presented method showed a very robust and flexible performance. Only a few cars got lost due to the procedure. In the worst case (Dataset 1) up to ten percent and in the best case (Dataset 3) not one got lost (Table 5.3). Nevertheless, a great effectiveness could be proven. The remaining final areas (after intersection), which are examined later, are in the range from 17 % to 11 % of the original image (Table 5.2). Also clear to see, in the same table, is that the most effective single step is 'Select Area' because by this method large homogeneous areas are removed.

Moreover, the algorithm is able to classify images of 20 cm and 13 cm resolution with the same parameter setting which can be seen by comparing the segmentation results of Datasets 4 and 5 (Figure 5.9e and 5.10e). In comparison to the subsequent gradient-based step, further features are its inherent color incorporation and the very fast processing speed.

In detail, the presented method is capable to remove road markings which is illustrated in Dataset 3, 4 and 5 (Figure 5.8e, 5.9e and 5.10e). Road markings are often responsible for false positive detections because a rectangular plane, similar to the contour of a car, is spanned by two parallel markings. Examples of false positives due to road markings can be seen in Figure 5.12a, 5.13a and 5.14a.

In general, the influence of shadow is a problematic issue. Shadow poses problems to the algorithm due to the lower contrast of these areas compared to sunny areas. A good example is the car line on the right side of the road in Figure 5.7. Especially dark cars parking close to the shadow area are classified as shadow and will be lost for the subsequent gradient classification step. Another example is available in Figure 5.9 and 5.10 where the car line on the right side in the center contains cars which are either very close to or completely in shadow areas. Again, especially dark cars are problematic due to the low contrast between car and shadow area. Of course, a possible solution would be to tune the parameter of the region-growing more sensitively but as a consequence, the classification result of normal illuminated areas would suffer from that decision.

All in all, the parameters of the whole segmentation approach have to be not adjusted for all datasets. The parameters of the anisometry step and the compactness step are set according to the longest vehicle line which is expected.

It is interesting to see that beside cars other remaining objects are mostly dormers (see Dataset 1 and 2 in Figure 5.6e and 5.7e). Among others, these objects are also difficult to classify for the following gradient-based detector (Figure 5.11a and 5.11c). Consequently, dormers have to be removed in a preceding step as done in the determination of the ground regions step. This is a case where the importance of the disparity maps becomes obvious but also shows that every single step of the strategy supports other steps.

6.1.4 Vehicle classification using gradients

The combination of HOG features and AdaBoost belongs to standard state-of-the-art object detection methods which has been proven several times for car detection (Chapter 2). In contrast to the application in this thesis other approaches rely on extensive training, e.g., online training or large training sets or sophisticated methods to automatically extend the training set. A problem is that the process of selecting the training data is often very in-transparent and the selected training data plays a crucial role for the detection result. However, the presented method in this work is just applied without much emphasis on tuning the data in order to get a slightly better result. Instead, the idea was to improve car detection with different boundary conditions which leads to the combination of several methods (see Figure 2.1).

The results here show the performance of such a simple detector, and it can be seen that the achieved result is not satisfying without support (Dataset 1–5 and Figure 5.11, 5.12, 5.13, 5.14). In Dataset 1 (Figure 5.11a) mostly dormers and elements of the façade cause false positive detections. False positive detections which are simpler to tackle are, for instance, in the middle of the road due to neighboring cars which present that typical edge on one side of the body. Additionally, four false negative detections are in the same dataset in the upper right car line. There is also no response in the corresponding confidence map (Figure 5.11b). An explanation for this phenomenon after the final weighted selection might be due to the missing space between these cars or due to two stronger detections to the left and the right side which does not leave space for a car in the middle.

Similarly, Dataset 2 shows also many false positives due to dormers or parts of the roof which seem to give a moderate confidence response due to at least one strong edge like that of the roof ridge (Figure 5.11c). Additionally, many elements of the façade are incorrectly detected as cars which becomes clear in the confidence map where large regions are yellow (Figure 5.11d). However, one good property of the detector is its capability of detecting cars close to the roof or even partly occluded ones. Furthermore, the result of Dataset 3 shows the great sensitivity of the detector to road markings which are a major reason for false positives in that case (Figure 5.12a). The same dataset lets us make a statement about the provided orientation tolerance of the detector. The maximum tolerance seems to be approximately $\pm 15^\circ$ because cars in the lower part of Figure 5.12a are correctly detected. In contrast, cars in the upper right part which are not detected show an orientation of 20° (clockwise with zero at the top). In addition, many false positives occur due to multi-detections. The trigger is often a certain spacing between cars, or the trailer of the truck which has a large rectangular shape of the width of a car. One solution to tackle the false positives due to the spacing could be a different parameter setting in the final weighted selection algorithm (Section 3.6.1).

Furthermore, a similar situation as seen in the previously shown result can also be observed in Dataset 4 and 5 (Figure 5.13a and 5.14a). Many false positives and false negatives are present despite the fact that only the vertical detector has been applied. The results of the confidence maps clearly show the lack of sophistication of the classifier (Figure 5.13b and 5.14b).

6.1.5 Discussion of optional sections

The mentioned methods in Section 3.7 and 3.8 are not sufficient when used on their own but interesting regarding the impact of the utilized features. Sometimes, a combination with other methods is required. Due to the fact that the two methods were not investigated in depth only a brief discussion is done. Further results concerning the color validation part are presented in Leister [2013], and the moving object incorporation part in Tuermer et al. [2011a].

Validation using vehicle background and color information

Common aerial images are usually 24 bit RGB color images (each channel 8 bit). A simple approach has been created trying to better incorporate color values. It is assumed that color features of a car candidate can be matched with the distribution of the color features from a large training set – in order to validate the candidate in the HSV color space. A conclusion that could be drawn from the experiment is that the V channel of the HSV color space is most helpful regarding the separation of car class from the other tested classes [Leister, 2013]. The V channel adjusts the brightness of the selected color but is not a color in itself.

The question whether a car can be reliable detected exclusively based on color features can be answered by other statistics (Figure 6.1): More than three-quarters of the newly registered cars world wide are strictly speaking not colored which is the case for black, gray and white (Figure 6.1d). Almost one-quarter of cars newly registered in the world are white, same as in Europe and in North America (Figure 6.1c and 6.1a, respectively). The exception is China where almost one-quarter of newly registered cars was black (Figure 6.1b). If the trend is continuously stable, color is a highly overrated feature. However, may be color can not be used to extract cars themselves but to exclude certain areas. The following hypothesis assumes that the color distribution of all registered cars is the same as the newly registered ones in 2012. For instance, green cars only occur very rarely (1 % world wide) but green trees are often a reason for false positives due to their manifold textures. In some scenarios it could be worth thinking about rejecting all green objects – as a consequence, the overall completeness of the detection is then maximum 99 % only. However, thereby many false positives could be avoided.

Actually, the presented car detection strategy already has the color feature incorporated due to the color segmentation step (Section 3.3.2) which leads to a significantly better result. However, the benefit of an additional integration of the color feature is still controversial. Recently published works show, on the one hand, the usage of color (color probability maps) [Kembhavi et al., 2011] and on the other hand, color is not used due to the fact that colored cars are in the minority [Kozempel, 2012].

Moving object incorporation

The detection of moving objects in video data, using the change of pixels from the current image to the subsequent one, is a strategy which has already pursued for a long time

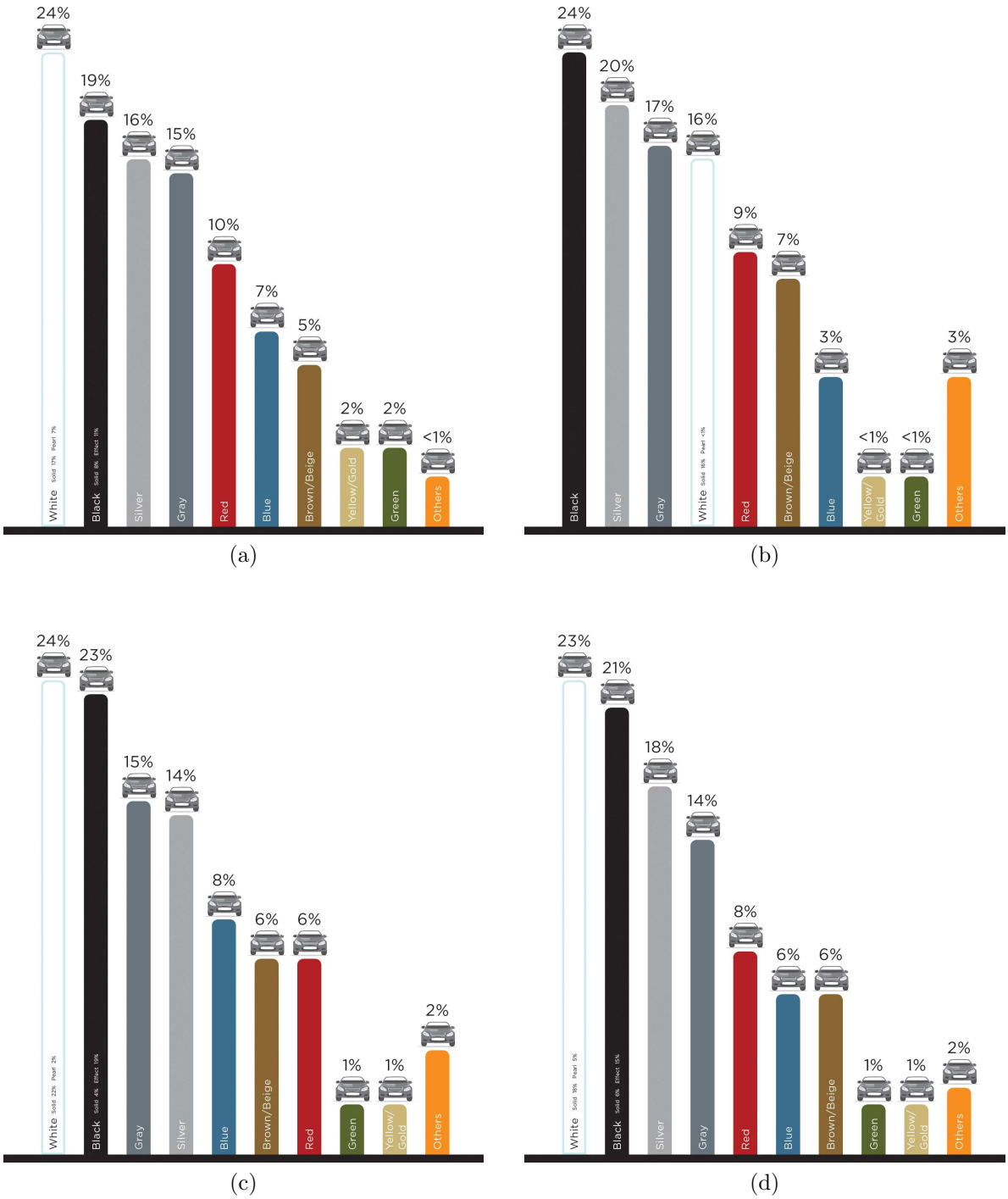


Figure 6.1: Automotive color popularity in the year 2012 [DuPont, 2012]. Top vehicle colors of (a) North America (b) China (c) Europe (d) World. The trend is still towards colors which are strictly speaking not a color such as black, gray and white. More than three quarters are in this group world wide (see (d)). This fact leads also to the conclusion that color is an overrated feature for detecting ordinary cars – but this statement is not valid for the separation of foreground and background. ©2012 DuPont

(Section 2.1). The same method is also applicable for image data recorded with a low frame-rate up to a certain limit (Section 2.2). Due to the simplicity and the robustness of the method, a very accurate detection result can be obtained. Potential inaccurate co-registration of the two images can be coped by adding a sliding window object detection technique [Tuermer et al., 2011a]. An improvement of the difference image technique in urban areas can be obtained with the integration of disparity maps.

However, only moving objects can be detected which excludes stationary and parked cars. In order to use the advantage of the detection of moving objects a solution could be to fuse the two detection methods. One method aims to detect all cars, but those moving cars are validated by the change detection approach. As result shown in [Tuermer et al., 2011a], the completeness of detected moving cars rises significantly.

6.2 Discussion of the complete car detection strategy

All steps of the presented strategy applied together are discussed in the following paragraphs.

The final result of Dataset 1 shows only two false positives which are due to the texture of the façade to the right of the road (Figure 5.15a). However, three cars are not detected in the upper part of the right car line. These false negatives are caused by the gradient based detector. They were already missing after the independent application of the single step (Section 5.1.4) which can be seen in Figure 5.11a. Moreover, this dataset is a good example to show the robustness of the classifier to slightly rotated cars. Please note, the rotation is not by chance but the dataset is cut out from a large scene according to the road segment of the Navteq database which is shown in Figure 5.1.

Similarly, the car lines in Dataset 2 are almost perpendicular to the upper boarder of the image. Five cars in the right car line are not detected (Figure 5.15b). However, the fault is only partly due to the gradient based detector. Some cars close to the shadow area, triggered by the house on the right, are already lost after the segmentation step (Figure 5.7e). A situation which has not happened as often as the wrong classification of the gradient based classifier.

Furthermore, the false positive detections in Dataset 3 are caused by objects on the construction site or by the grassed area (Figure 5.16). Avoiding the false positives on the construction site is not easy due to their car-like contour and texture. If possible, an efficient solution would be the restriction of the search area to roads only. False positives on the grassed area can be avoided by using color statistics (Section 6.1.5). Moreover, it can be realized that trucks are not detected. In this work, trucks were not considered as target objects and thus the model of the strategy does not include trucks. Also not detected are cars in the upper right car line and in the center left car line which might be due to their orientation and the non-invariant gradient-based detector. A positive aspect is that also partly occluded cars are detected (e.g., occlusion by the tower crane in the lower part of the image).

The large datasets (Dataset 4 and 5) show all problematic situations similar to the occurrence in the other datasets such as façades, rectangular objects the size of a car and shadow areas (Figure 5.17). However, due to the fact that detectors of two orientations were applied the result is not directly comparable to Dataset 1 to 3.

The completeness-correctness graph illustrated in Figure 5.18 gives an impression of the performance of the strategy related to the complexity of the dataset concerning car detection. Dataset 1 can be considered as easy for the presented strategy, and the completeness and correctness values are often at a high level ($> 90\,\%$). The corresponding quality is at $82\,\%$ in Table 5.4. In contrast, Dataset 5, which has large ground level areas, shows a bad result - the curve in Figure 5.18 is dropping much faster. Also the quality is only at $64\,\%$ in Table 5.4 for that dataset. In other works, benchmark numbers are often calculated when cars are only detected on roads which are precisely masked out using an accurately fitting road mask from databases. A case which can hardly be compared.

For all datasets, the following applies.

- The utilized disparity maps enhance the overall detection result and are supportive of the other strategy steps, especially in densely populated urban areas.
- In general, segmentation is a crucial step but has been shown a robustness to different sensors and to slightly changing spatial resolutions. The benefit can be expressed by the enabled use of the boosting method utilized here with low training effort. A drawback of the RGB color region-growing is that shadow areas are often determined as one region. Another color space might provide a solution (e.g., Lab color space), at least for colored cars. Then the separation between ground (mostly asphalt) in shadow areas and cars in shadow areas is easier.
- Furthermore, the combination of disparity maps and segmentation turned out to be a good strategy which does not lead to redundant information. Dormers, for instance, can only be removed by the disparity maps. In contrast, road markings can only be removed by the segmentation step.
- In addition, the final weighted selection step is also robust. But of course the size of the rectangle should not be too large in order to preserve all cars instead of losing ones which are parked close to each other. However, this parameter can be easily adjusted depending on the resolution of the data. Finally, the detected vehicles could be refined using the grouping of vehicles but single cars are then discriminated. Also a potential method is the CRF which introduces relations to neighboring segments. Vehicle queues can be easily incorporated in the final weighted selection. The idea was introduced a while ago but the impact of this method might depend on several issues [Burlina et al., 1997]. Exploiting contextual knowledge of parked cars has been also done by Stilla & Michaelsen [2002] and Leitloff et al. [2010].

7 Conclusion and Outlook

The next section concludes this thesis and the outlook section gives hints for potential further improvement of car detection in aerial imagery.

7.1 Conclusion

In this work a strategy for vehicle detection in aerial imagery has been presented. Different info sources (maps, images) and descriptive features were exploited in order to achieve high quality results. Vehicle detection in aerial imagery is more than the development of a single detector using the latest object detection approach. This starts with the appropriate use of previous knowledge from road databases. A task which has to be done according to the location of the area of interest and to the objective for which the cars are extracted. In case of the presented strategy, which aims to detect all cars in dense urban areas, road segments from databases are only used for an approximate limitation of the area of interest.

Moreover, the strategy is also based on real-time disparity images which are calculated directly before extracting the cars. A method that showed its excellent suitability in densely populated urban areas. Especially, urban canyons provide a very good scenario to illustrate the high effectiveness of disparity maps. Many objects, such as dormers or elements of façades, which are sometimes recognized as cars, could be successfully excluded. In contrast, flat areas, for instance, in rural regions need a different kind of treatment because the disparity maps from the utilized test data are not applicable for the fine distinction of objects with a height difference of only a few decimeters.

Furthermore, an essential part of the strategy is the segmentation and the rotation invariant extraction of candidate regions. The most important step is the smoothing in combination with the preservation of certain edges such as the contours of the main body of the cars. Subsequently, the color segmentation can be carried out on RGB images. After that, returned regions are filtered according to their geometric properties. The algorithm has been proven to have a very robust performance – in the worst case 11 % of the cars in the image were lost. However, in the same case the search area could be restricted to 16 % of the original image. In addition, the parameter setting of the method is simple but a high generality related to different resolutions is still present. Images of two sensors and with a different resolution could be processed without adjusting the parameter settings. Due to the segmentation strategy many objects like road markings or bike lanes could be removed. These objects have two strong parallel edges similar to the width of a car and thus they lead often to false positive detections. The reason is

that many cars, especially black ones, only appear as dark rectangular objects, and two parallel edges in a certain distance are then a reasonable description of a car.

However, the major reason for the importance of the segmentation and the disparity image is because they pave the way for the application of an loosely trained sliding window approach using HOG features. The combination of HOG features and AdaBoost has already shown its ability to classify cars but the training needs a lot of manual interaction – a point which can be abstained from when using the presented strategy. Even though, the application of the boosted classifier is not a key factor and can be easily substituted by another classifier.

The overall objective of the work, the presentation of a robust car detection approach, showed in the best case a completeness of 86 % and a correctness of 92 % at the same time. Although limitations of the approach became obvious during the tests. The quality of the classification result was reduced due to confusing elements from a construction site. These elements could not be excluded because of the mostly flat area.

A further problem of the strategy is the rotation variance of the gradient-based vehicle detector which leads to false negatives and reduces the completeness rate of the detection result. In addition, progress has also to be made in the treatment of shadow areas for which concepts have already been presented. Undoubtedly, the weakest link in the processing chain is the insufficiently trained gradient based detector. However, the power of these detectors is already proven under the condition of an intensive training. The best case scenario is the detector which is still robust to changing resolutions and which provides good results with a low training effort, i.e., preparation of the inertial training data and the iterative enhancement of the classifier by back-porting of wrongly classified objects.

7.2 Outlook

The continuous technical development of airborne platforms provides innovative opportunities. For several years vehicle detection in aerial images has been restricted to images which are taken from aircraft [Voss & Grüber, 2003]. However, since small UAVs / RPAS (Remotely Piloted Aircraft Systems) are becoming cheaper and more popular, more and more work is carried out on images taken from RPAS [Moranduzzo & Melgani, 2012, 2013]. The advantage remaining of images from aircraft is a wide coverage of the target area. In contrast, RPAS are restricted to a shorter operating distance (e.g., range of sight) and a shorter operation time. Nevertheless, the GSD of the images is many times higher and an GSD of up to 2 cm can be expected.

Consequently, due to this high resolution images other state of the art methods or their modifications can be applied. It is reasonable to try the method of deformable parts (DPM) by Felzenszwalb et al. [2010], for instance, when the high resolution allows recognition of typical single car parts. Moreover, spatial pyramid matching (SPM) by Lazebnik et al. [2006] could be applied. A method that has also been combined with sparse codes (SC) [Yang et al., 2009] and offers future potential for car detection in high-resolution imagery. An extension of sparse representations for object detection are the histograms of sparse codes (HSC) [Ren & Ramanan, 2013]. Furthermore, an extension of the method

known as shape context is named feature context (FC) and is presented by Wang et al. [2011]. It has also not yet been applied to aerial images for car detection.

The following additional aspects may be mentioned in order to improve car detection in aerial imagery:

- Context is still important when aiming to extract cars from aerial imagery. A method that carries out an iterative segmentation plus detection is shown by Sun et al. [2012]. The major focus is on the contextual relationship between objects and the scene geometric. Also the road surface is related to context which could be really helpful for some car detection applications. There are several automatic road detection systems. Recently, one has been developed for high-resolution aerial images [Mnih & Hinton, 2010].
- The extension of the AdaBoost detector with additional features like Haar-like, LBP, Gabor, SIFT, SURF was not realized. This work was already done at an earlier time (see Chapter 2). An enhancement by doing so could be expected but still has to be proven. Moreover, another possible improvement is the application of another machine learning technique. However, in my opinion the results would be only slightly better because the major impact is related to the utilized features and the selected training data.
- Other works in the field of vehicle detection do not yet consider the direction of the gradient vector. The idea is to separate bright cars from dark cars and create two different detectors because the gradients of dark objects on a bright background is oriented in the opposite orientation than a bright object on a dark background. Unfortunately, preliminary tests did not return successful results.
- As discussed in Section 6.1.3, an alternative solution could be the determination of shadow areas [Makarau et al., 2011; Das & Aery, 2013]. However, the determination of such areas by only using the image is not easy as own investigations showed. A comparison of selected algorithms which are applied to a simple scene including cars is shown by Chung et al. [2009]. Due to the limited information dark objects are often classified as shadow areas which also includes dark cars. Hence an approach using a geometric solution by integrating a DSM and the position of the sun can lead to better results [Li et al., 2005]. Finally, after the determination of the shadow areas the histogram of these areas can be adjusted in several ways in order to equalize shady and sunny areas. An approach for very high resolution images is presented by Lorenzi et al. [2013] where also fine structures in shady areas could be restored and their contrast gets equal to normal illuminated areas. However, the question whether dark objects such as black cars can be always enhanced in the same way remains unanswered by this article.

Bibliography

AdV (1996) Amtliches Topographisch-Kartographisches Informationssystem (ATKIS). http://www.adv-online.de/. (accessed 26.11.2012).

Agamennoni G, Nieto JI, Nebot EM (2010) Robust and Accurate Road Map Inference. In: *IEEE International Conference on Robotics and Automation (ICRA)*

Albrecht U, Heimann J, Schulz W (1995) Method and system for the prognosis of the traffic stream. European Patent EP0755039A2. Mannesmann AG.

Aubert G, Kornprobst P (2006) *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, volume 147 of *Applied Mathematical Sciences*. New-York: Springer, New-York, second edition.

Banister D, Browne M, Givonia M (2010) Transport Reviews - The 30th Anniversary of the Journal. *Transport Reviews: A Transnational Transdisciplinary Journal*, 30: 1–10.

Bauer E, Kohavi R (2009) An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. *Machine Learning*, 36 (1–38): 105–139.

Baumgartner S, Krieger G (2011) Fast GMTI Algorithm For Traffic Monitoring Based On A Priori Knowledge. *IEEE Transactions on Geoscience and Remote Sensing*, 50 (11): 4626–4641.

Bay H, Ess A, Tuytelaars T, Gool LV (2008) SURF Speeded Up Robust Features. *Computer Vision and Image Understanding (CVIU)*, 110 (3): 346–359.

Bhattacharyya A (1943) On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society*, 35: 99–109.

Bischof H, Godec M, Leistner C, Rinner B, Starzacher A (2010) Autonomous Audio-Supported Learning of Visual Classifiers for Traffic Monitoring. *IEEE Intelligent Systems*, 25 (3): 15–23.

Breckon TP, Barnes SE, Eichner ML, Wahren K (2008) Autonomous Real-time Vehicle Detection from a Medium-Level UAV. In: *International Unmanned Air Vehicle Systems Conference*

Breiman L, Friedman J, Olshen R, Stone C (1984) *Classification and Regression Trees*. Boca Raton - London - New York - Washington D.C.: Chapman & Hall/CRC.

Brenner AR, Essen H, Stilla U (2012) Representation of stationary vehicles in ultra-high resolution SAR and turntable ISAR images. In: *European Conference on Synthetic Aperture Radar (EUSAR)*: 147–150.

Burlina P, Parameswaran V, Chellappa R (1997) Sensitivity Analysis and Learning Strategies for Context-Based Vehicle Detection Algorithms. In: *DARPA IU Workshop 97*

Busch F, Glas F, Bermann E (2004) Dispositionssysteme als FCD-Quellen für eine verbesserte Verkehrslagerekonstruktion in Städten. *Straßenverkehrstechnik*, 4: 437–444.

Canavosio-Zuzelski R (2013) *A Photogrammetric Approach for Geopositioning OpenStreetMap Roads*. PhD thesis, George Mason University, Fairfax, VA, USA.

Canavosio-Zuzelski R, Agouris P, Doucette P (2013) A Photogrammetric Approach for Assessing Positional Accuracy of OpenStreetMap Roads. *ISPRS International Journal of GeoInformation*, 2: 276–301.

Cao X, Lan J, Yan P, Li X (2011a) KLT Feature Based Vehicle Detection and Tracking in Airborne Videos. In: *International Conference on Image and Graphics (ICIG)*: 673–678.

Cao X, Lan J, Yan P, Li X (2012a) Vehicle detection and tracking in airborne videos by multimotion layer analysis. *Machine Vision and Applications*, 23: 921–935.

Cao X, Lin R, Yan P, Li X (2012b) Visual Attention Accelerated Vehicle Detection in Low-Altitude Airborne Video of Urban Environment. *IEEE Transactions on Circuits and Systems for Video Technology*, 22 (3): 366–378.

Cao X, Wu C, Lan J, Yan P, Li X (2011b) Vehicle Detection and Motion Analysis in Low-Altitude Airborne Video Under Urban Environment. *IEEE Transactions on Circuits and Systems for Video Technology*, 21 (10): 1522–1533.

Cao X, Wu C, Yan P, Li X (2011c) Linear SVM classification using boosting HOG features for vehicle detection in low-altitude airborne videos. In: *IEEE International Conference on Image Processing (ICIP)*: 2421–2424.

Casasent DP, Chen XW (2003) Mine and vehicle detection in hyperspectral image data: waveband selection. In: Sadjadi FA (ed) *Automatic Target Recognition XIII*, SPIE 5094: 228–241.

Cerutti-Maori D, Klare J, Brenner AR, Ender JHG (2008) Wide-Area Traffic Monitoring With the SAR/GMTI System PAMIR. *IEEE Transactions on Geoscience and Remote Sensing*, 46 (10): 3019–3030.

Chandrasekhar V, Takacs G, Chen D, Tsai S, Grzeszczuk R, Girod B (2009) CHoG: Compressed histogram of gradients A low bit-rate feature descriptor. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*: 2504–2511.

Chandrasekhar V, Takacs G, Chen DM, Tsai SS, Reznik Y, Grzeszczuk R, Girod B (2012) Compressed Histogram of Gradients: A Low-Bitrate Descriptor. *International Journal of Computer Vision*, 96 (3): 384–399.

Chang P, Krumm J (1999) Object recognition with color cooccurrence histograms. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*

Chang WC, Cho CW (2010) Online Boosting for Vehicle Detection. *IEEE Transactions on Systems, Man, and Cybernetics*, 40 (3): 892–902.

Chen YG, Giga Y, Goto S (1991) Uniqueness and existence of viscosity solutions of generalized mean curvature flow equations. *Journal of Differential Geometry*, 33: 749–786.

Cheng HY, Weng CC, Chen YY (2012) Vehicle Detection in Aerial Surveillance Using Dynamic Bayesian Networks. *IEEE Transactions on Image Processing*, 21 (4): 2152–2159.

Choi JY, Yang YK (2009) Vehicle Detection from Aerial Images Using Local Shape Information. In: *Pacific Rim Symposium on Advances in Image and Video Technology (PSIVT)*: 227–236.

Chung KL, Lin YR, Huang YH (2009) Efficient Shadow Detection of Color Aerial Images Based on Successive Thresholding Scheme. *IEEE Transactions on Geoscience and Remote Sensing*, 47 (2): 671–682.

Clarenz U, Dziuk G, Rumpf M (2003) *Geometric Analysis and Nonlinear Partial Differential Equations*. Springer-Verlag Berlin Heidelberg New York. Editors: Hildebrandt S, Karcher H.

Clark MAG (1983) Induction loop vehicle detector. US Patent US4568937A. Microsense Systems, Limited.

Comaniciu D, Ramesh V, Meer P (2003) Kernel-Based Object Tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 25 (5): 564–577.

Cramer M (2010) The DGPF - Test on Digital Airborne Camera Evaluation – Overview and Test Design. *Photogrammetrie - Fernerkundung - Geoinformation (PFG)*, 2010 (2): 73–82.

Crandall MG, Lions PL (1996) Convergent difference schemes for nonlinear parabolic equations and mean curvature motion. *Numerische Mathematik*, 75: 17–41.

Dalal N (2006) *Finding People in Images and Videos*. PhD thesis, Institut National Polytechnique de Grenoble.

Dalal N, Triggs B (2005) Histograms of Oriented Gradients for Human Detection. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1: 886 – 893.

d'Angelo P, Reinartz P (2011) Semiglobal Matching Results on the ISPRS Stereo Matching Benchmark. In: *ISPRS Hannover Workshop - High-Resolution Earth Imaging for Geospatial Information*

Das S, Aery A (2013) A Review: Shadow Detection And Shadow Removal from Images. *International Journal of Engineering Trends and Technology (IJETT)*, 4 (5): 1764–1767.

Davidson L, Valentine E (2001) Wireless induction loop control system. US Patent US6265788B1. Ericsson Inc.

Dubuisson MP, Jain AK (1995) Contour extraction of moving objects in complex outdoor scenes. *International Journal of Computer Vision*, 14 (1): 83–105.

DuPont (2012) *DuPont 2012 Global Automotive Color Popularity Report*. E. I. du Pont de Nemours and Company, Technical report.

Eikvil L, Aurdal L, Koren H (2009) Classification-based vehicle detection in high-resolution satellite images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64: 65–72.

Ender JHG, Gierull CH, Cerutti-Maori D (2008) Improved Space-Based Moving Target Indication via Alternate Transmission and Receiver Switching. *IEEE Transactions on Geoscience and Remote Sensing*, 46 (12): 3960–3974.

Felzenszwalb PF, Girshick RB, McAllester D, Ramanan D (2010) Object Detection with Discriminatively Trained Part-Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32 (9): 1627–1645.

Fischler MA, Bolles RC (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24 (6): 381–395.

Frei W, Chen C (1977) Fast boundary detection: A generalization and new algorithm. *IEEE Transactions on Computers*, C-26: 988–998.

Freund Y (1990) Boosting a weak learning algorithm by majority. In: *Third Annual Workshop on Computational Learning Theory*

Freund Y, Schapire R (1995) A decision-theoretic generalization of on-line learning and an application to boosting. In: *Proceedings of the Second European Conference on Computational Learning Theory*: 23–37.

Freund Y, Schapire RE (1996) Experiments with a New Boosting Algorithm. In: *Machine Learning: Proceedings of the Thirteenth International Conference*

Freund Y, Schapire RE (1997) A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55 (1): 119–139.

Friedman J, Hastie T, Tibshirani R (2000) Additive logistic regression: A statistical view of boosting. *Annals of Statistics*, 28 (2): 337–374.

Fukunaga K, Hostetler LD (1975) The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition. *IEEE Transactions on Information Theory*, 21 (1): 32–40.

Förstner W, Gülch E (1987) A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centers of Circular Features. In: *ISPRS Intercommission Workshop on Fast Processing of Photogrammetric Data*: 281–305.

Gerke M, Heipke C (2008) Image based quality assessment of road databases. *International Journal of Geoinformation Science*, 22 (8): 871–894.

Givoni I, Li P, Frey B (2011) Learning Better Image Representations Using 'Flobject Analysis'. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*: 2721–2728.

Gleason J, Nefian AV, Bouyssounousse X, Fong T, Bebis G (2011) Vehicle Detection from Aerial Imagery. In: *IEEE International Conference on Robotics and Automation (ICRA)*

Gomes A, Voiculescu I, Jorge J, Wyvill B, Galbraith C (2009) *Implicit Curves and Surfaces: Mathematics, Data Structures and Algorithms*, volume XIV. Springer London.

Gonzalez RC, Woods RE (2007) *Digital Image Processing*. Prentice Hall, 3 edition.

Google (2009) Google Maps for mobile. http://googleblog.blogspot.de/2009/08/bright-side-of-sitting-in-traffic.html (accessed 21/08/2013).

Gorokhovich Y, Voustianiouk A (2006) Accuracy assessment of the processed SRTM-based elevation data by CGIAR using field data from USA and Thailand and its relation to the terrain characteristics. *Remote Sensing of Environment*, 104 (2006): 409–415.

Grabner H (2008) *On-line Boosting and Vision*. PhD thesis, TU Graz.

Grabner H, Nguyen TT, Gruber B, Bischof H (2008) On-line boosting-based car detection from aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 63 (3): 382–396.

Haag M, Nagel HH (1999) Combination of Edge Element and Optical Flow Estimates for 3D-Model-Based Vehicle Tracking in Traffic Image Sequences. *International Journal of Computer Vision*, 35 (3): 295–319.

Haklay M (2010) How good is volunteered geographical information? A comparative study of OpenStreetMap and Ordnance Survey datasets. *Environment and Planning B: Planning and Design*, 37 (4): 682–703.

Hamming RW (1950) Error detecting and error correcting codes. *Bell System Technical Journal*, 29 (2): 147–160.

Harris C, Stephens M (1988) A combined corner and edge detector. In: *Proceedings of the 4th Alvey Vision Conference*: 147–151.

Hartley R, Zisserman A (2010) *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, 2 edition.

Hausburg M (2010) Validierung von Objekthypothesen in ARGOS-Luftbildern. Master's thesis, Technische Universität Berlin.

Hechenbichler K (2005) *Ensemble-Techniken und ordinale Klassifikation*. PhD thesis, Ludwig-Maximilians-Universität (LMU) München.

Heinrichs M (2011) *Automatische Generierung von 3D-Modellen mittels Sequenzen hochauflösender Bildtripel*. PhD thesis, Technischen Universität Berlin.

Heitz G, Koller D (2008) Learning Spatial Context: Using Stuff to Find Things. In: *European Conference on Computer Vision (ECCV)*: 30–43.

Hinsbergen Cv (2010) *Bayesian Data Asimilation for Improved Modelling of Road Traffic*. PhD thesis, TRAIL Research School, Netherlands.

Hinz S (2004) Detection of Vehicles and Vehicle Queues in High Resolution Aerial Images. *Photogrammetrie - Fernerkundung - Geoinformation (PFG)*, 3/04: 201–213.

Hinz S, Bamler R, Stilla U (2006) Editorial Theme Issue: Airborne und Spaceborne Traffic Monitoring. *ISPRS Journal of Photogrammetry and Remote Sensing*, 61 (3-4): 135–136.

Hinz S, Stilla U (2006) Car detection in aerial thermal images by local and global evidence accumulation. *Pattern Recognition Letters*, 27 (4): 308–315.

Hirschmueller H (2008) Stereo Processing by Semi-Global Matching and Mutual Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 30 (2): 328–341.

Hirschmueller H, Scharstein D (2009) Evaluation of stereo matching costs on image with radiometric differences. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 31 (9): 1582–1599.

Hoffmann J, Walter D (2006) How Complementary are SRTM-X and -C Band Digital Elevation Models? *Photogrammetric Engineering and Remote Sensing*, 72 (3): 261–268.

Holt AC, Seto EYW, Rivard T, Peng G (2009) Object-based Detection and Classification of Vehicles from High-resolution Aerial Photography. *Photogrammetric Engineering and Remote Sensing*, 75 (7): 871–880.

Huang CC, Wang SJ (2010) A Hierarchical Bayesian Generation Framework for Vacant Parking Space Detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 20 (12): 1770–1785.

Jin X, Davis CH (2007) Vehicle detection from high-resolution satellite imagery using morphological shared-weight neural networks. *Image and Vision Computing*, 25 (9): 1422–1431.

Kahmen H (2005) *Angewandte Geodäsie: Vermessungskunde*. De Gruyter Lehrbuch. Walter de Gruyter, 20 edition.

Kailath T (1967) The Divergence and Bhattacharyya Distance Measures in Signal Selection. *IEEE Transactions on Communication Technology*, 15 (1): 52–60.

Kasturi R, Goldgof D, Soundararajan P, Manohar V, Garofolo J, Bowers R, Boonstra M, Korzhova V, Zhang J (2009) Framework for Performance Evaluation of Face, Text, and Vehicle Detection and Tracking in Video: Data, Metrics, and Protocol. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 31 (2): 319–336.

Kembhavi A, Harwood D, Davis L (2011) Vehicle Detection Using Partial Least Squares. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 33 (6): 1250–1265.

Khan S, Cheng H, Matthies D, Sawhney H (2010) 3D Model Based Vehicle Classification in Aerial Imagery. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*: 1681–1687.

Kienzle J (2001) Analyse von Einzelfahrzeugdaten - Verkehr verstehen. Master's thesis, Universität Stuttgart.

Kim Z, Malik J (2003) Fast vehicle detection with probabilistic feature grouping and its application to vehicle tracking. In: *International Conference on Computer Vision (ICCV)*: 524–531.

Kirchhof M, Stilla U (2006) Detection of moving objects in airborne thermal videos. *ISPRS Journal of Photogrammetry and Remote Sensing*, 61 (Issues 3-4): 187–196.

Kirsch R (1971) Computer determination of the constituent structure of biological images. *Computers and Biomedical Research*, 4: 315–328.

Kittler J, Illingworth J (1986) Minimum error thresholding. *Pattern Recognition*, 19: 41–47.

Kluckner S (2011) *Semantic Interpretation of Digital Aerial Images Utilizing Redundancy, Appearance and 3D Information*. PhD thesis, Graz University of Technology.

Kluckner S, Pacher G, Grabner H, Bischof H (2007) A 3D Teacher for Car Detection in Aerial Images. In: *International Conference on Computer Vision (ICCV)*

Knauer U, Reulke R, Meffert B (2005) Fahrzeugdetektion und -erkennung mittels mehrdimensionaler Farbhistogrammanalyse. In: *Farbbildverarbeitung 2005*: 93–100.

Kozempel K (2012) *Entwicklung und Validierung eines Gesamtsystems zur Verkehrserfassung basierend auf Luftbildsequenzen*. PhD thesis, Humboldt-Universität zu Berlin.

Kozempel K, Reulke R (2009) Fast Vehicle Detection and Tracking in Aerial Image Bursts. In: Stilla U, Rottensteiner F, Paparoditis N (eds) *City Models, Roads and Traffic (CMRT)*, 38 (3/W4): 175–180.

Kraftfahrt-Bundesamt (2011) *Fachartikel: Farbe der Fahrzeuge*. Kraftfahrt-Bundesamt, Referat Fahrzeugstatistik, Sachgebiet 321, Technical report.

Kraftfahrt-Bundesamt (2012) *Neuzulassungen von Personenkraftwagen im Jahr 2011 nach Farben*. Statistische Mitteilungen des Kraftfahrt-Bundesamtes, Technical report.

Kraus K (2007) *Photogrammetry - Geometry from Images and Laser Scans*. Walter de Gruyter, Berlin, 2 edition.

Kroon D (2009) *Numerical Optimization of Kernel Based Image Derivatives*. University Twente, Technical report.

Kurz F, Müller R, Stephani M, Reinartz P, Schroeder M (2007) Calibration of a Wide-Angel Digital Camera System for Near Real Time Scenarios. In: Heipke C, Jacobsen K, Gerke M (eds) *ISPRS Hannover Workshop - High-Resolution Earth Imaging for Geospatial Information*

Kurz F, Türmer S, Meynberg O, Rosenbaum D, Leitloff J, Runge H, Reinartz P (2012) Low-cost optical camera systems for real time mapping applications. *Photogrammetrie - Fernerkundung - Geoinformation (PFG)*, 2012 (2): 159–175.

Laparmonpinyo P, Chitsobhuk O (2010) A video-based traffic monitoring system based on the novel gradient-edge and detectionwindow techniques. In: *Computer and Automation Engineering (ICCAE)*: 30–34.

Larsen SO, Koren H, Solberg R (2009) Traffic Monitoring Using Very High Resolution Satellite Imagery. *Photogrammetric Engineering and Remote Sensing*, 75 (7): 859–869.

Lazebnik S, Schmid C, Ponce J (2006) Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*: 2169–2178.

Leberl F, Bischof H, Grabner H, Kluckner S (2007) Recognizing Cars in Aerial Imagery to Improve Orthophotos. In: *Proceedings ACM International Symposium on Advances in Geographic Information Systems*

Leberl F, Kluckner S, Pacher G, Grabner H, Bischof H, Gruber M (2008) Detecting cars in aerial imagery for improvements of orthophotos and digital elevation models. In: *ASPRS Annual Conference*

Lee YJ, Yilmaz A (2011) Boresight Calibration of the Aerial Multi-Head Camera System. In: Blowers M, O'Donnell TH, Mendoza-Schrock OL (eds) *Evolutionary and Bio-Inspired Computation: Theory and Applications V*: Orlando, Florida, USA, SPIE 8059.

Leibe B, Leonardis A, Schiele B (2008) Robust Object Detection with Interleaved Categorization and Segmentation. *International Journal of Computer Vision*, 77 (1): 259–289.

Leister W (2013) Hypothesenvalidierung von Fahrzeugdetektionen mit Hilfe von Color-Cooccurrence-Histogrammen im HSV-Farbraum. Master's thesis, Technische Universität Chemnitz.

Leister W, Tuermer S, Reinartz P, Hoffmann KH, Stilla U (2013) Validation of vehicle candidate areas in aerial images using color co-occurrence histograms. In: *ISPRS Conference on Serving Society with Geoinformatics (SSG)*: Antalya, Turkey, XL-7/W2: 139–144.

Leitloff J (2011) *Detektion von Fahrzeugen in optischen Satellitenbildern*. PhD thesis, Technische Universität München (TUM).

Leitloff J, Hinz S, Stilla U (2010) Vehicle extraction from very high resolution satellite images of city areas. *IEEE Transactions on Geoscience and Remote Sensing*, 48 (7): 2795–2806.

Lenhart D, Hinz S (2006) Automatic Vehicle Tracking in Low Frame Rate Aerial Image Sequences. In: *International Archives of Photogrammetry, Remote Sensing, and Spatial Information Sciences*, 36 (3): 203–208.

Lenhart D, Hinz S, Leitloff J, Stilla U (2008) Automatic traffic monitoring based on aerial image sequences. *Pattern Recognition and Image Analysis*, 18 (3): 400–405.

Leonhardt AJ (2008) *Ein Instanzbasiertes Lernverfahren zur Prognose von Verkehrskenngrößen unter Nutzung räumlich-zeitlicher Verkehrsmuster*. PhD thesis, Technische Universität München.

Leutenegger S, Chli M, Siegwart R (2011) BRISK: Binary Robust Invariant Scalable Keypoints. In: *International Conference on Computer Vision (ICCV)*

Li S, Zhang B, Gao L, Sun X (2009) Small objects detection of hyperspectral image in urban areas. In: *Joint Urban Remote Sensing Event (JURSE)*

Li Y, Gong P, Sasagawa T (2005) An Image Analysis and Photogrammetric Engineering Integrated Shadow Detection Model. In: *Developments in Spatial Data Handling*: 547–557.

Lienhart R, Kuranov A, Pisarevsky V (2003) Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection. In: *Pattern Recognition*, volume 2781 of *Lecture Notes in Computer Science* (pp. 297–304). Springer.

Lim KH, Ang LM, Seng KP, Chin SW (2009) Lane-Vehicle Detection and Tracking. In: *International MultiConference of Engineers and Computer Scientists*, 2.

Liu L, Fieguth P, Clausi D, Kuang G (2012) Sorted random projections for robust rotation-invariant texture classification. *Pattern Recognition*, 45 (6): 2405–2418.

Lorenzi L, Melgani F, Mercier G, Bazi Y (2013) Assessing the Reconstructability of Shadow Areas in VHR Images. *IEEE Transactions on Geoscience and Remote Sensing*, 51 (5): 2863–2873.

Lowe DG (1999) Object recognition from local scale-invariant features. In: *International Conference on Computer Vision (ICCV)*, 2: 1150–1157.

Lowe DG (2004) Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60 (2): 91–110.

Ludwig I, Voss A, Krause-Traudes M (2011) A comparison of the street networks of Navteq and OSM in Germany. In: *International Conference on Geographic Information Science (AGILE)*: 65–84.

Lv W, Jiang X, Li C, Zhu T (2011) A Novel Model to Evaluate Urban Overall Traffic Condition. In: *International Conference on ITS Telecommunications (ITST)*: 515–520.

Makarau A, Richter R, Müller R, Reinartz P (2011) Adaptive Shadow Detection Using a Blackbody Radiator Model. *IEEE Transactions on Geoscience and Remote Sensing*, 49 (6): 2049–2059.

Maksymiuk O, Schmitt M, Brenner AR, Stilla U (2012) First investigations on detection of stationary vehicles in airborne decimeter resolution sar data by supervised learning. In: *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*: 3584–3587.

Manolakis D, Marden D, Shaw GA (2003) Hyperspectral Image Processing for Automatic Target Detection Applications. *Lincoln Laboratory Journal*, 14 (1): 79–116.

Matsur IY (2011) Traffic monitoring system. US Patent US0246210A1.

Mauthner T, Kluckner S, Roth P, Bischof H (2010) Efficient Object Detection Using Orthogonal NMF Descriptor Hierarchies. In: *Annual Symposium German Association for Pattern Recognition*: 212–221.

Meng L, Kerekes J (2012) Object tracking using high resolution satellite imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS)*, 5 (1): 146–152.

Mikolajczyk K, Schmid C (2005) A Performance Evaluation of Local Descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 27 (10): 1615–1630.

Mirchandani P, Hickman M, Angel A, Chandnani D (2002) Application of Aerial Video for Traffic Flow Monitoring and Management. In: *Integrating Remote Sensing at the Global, Regional and Local Scale. Pecora 15/Land Satellite Information IV Conference*

Mnih V, Hinton GE (2010) Learning to Detect Roads in High-Resolution Aerial Images. In: *European Conference on Computer Vision (ECCV)*, 6316: 210–223.

Moon H, Chellappa R, Rosenfeld A (2002) Performance Analysis of a Simple Vehicle Detection Algorithm. *Image and Vision Computing*, 20 (1): 1–13.

Moranduzzo T, Melgani F (2012) A SIFT-SVM method for detecting cars in UAV images. In: *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*: 6868–6871.

Moranduzzo T, Melgani F (2013) Automatic Car Counting Method for Unmanned Aerial Vehicle Images. *IEEE Transactions on Geoscience and Remote Sensing*, PP: 1–13. (online availible).

Murphy K, Torralba A, Eaton D, Freeman WT (2006) Object detection and localization using local and global features. *Lecture Notes in Computer Science*, 4170: 382–400. Towards Category-Level Object Recognition.

MVTec (2012) *HALCON Operator Reference*. MVTec Software GmbH, 11 edition.

NAVTEQ (1985) Subsidiary of Nokia. http://corporate.navteq.com. (accessed 19/11/2012).

NAVTEQ (2010) Building the NAVTEQ Map. http://press-de.navteq.com/download /Building+the+NAVTEQ+Database.pdf. (accessed 22/08/2013).

Negri P, Clady X, Hanif SM, Prevost L (2008) A cascade of boosted generative and discriminative classifiers for vehicle detection. *EURASIP Journal on Advances in Signal Processing*, 2008: 1–12.

Nejadasl FK (2005) Automatic traffic monitoring by helicopter video imagery with optical flow method. *Leonardo Times*, 4: 30–32.

Nejadasl FK (2010) *A System for the Acquisition and Analysis of Image Sequences to Model Longitudinal Driving Behavior*. PhD thesis, Technische Universiteit Delft.

Nejadasl FK, Gorte BG, Hoogendoorn SP (2006) Optical flow based vehicle tracking strengthened by statistical decisions. *ISPRS Journal of Photogrammetry and Remote Sensing*, 61 (3-4): 159–169. Theme Issue: Airborne and Spaceborne Traffic Monitoring.

Nejadasl FK, Lindenbergh R (2011) Automatic traffic monitoring by helicopter video imagery. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science*: Munich, Germany, 38, part 3 / W22.

Nguyen TT, Grabner H, Gruber B, Bischof H (2006) On-line Boosting for Car Detection from Aerial Images. In: *IEEE International Conference on Computer Sciences (RIVF)*

Nurul Habib KM, Morency C, Trépanier M (2012) Integrating parking behaviour in activity-based travel demand modelling: Investigation of the relationship between parking type choice and activity scheduling process. *Transportation Research Part A*, 46 (1): 154–166.

OpenStreetMap (2004) http://www.openstreetmap.org. (accessed 19/11/2012).

Pacher G, Kluckner S, Bischof H (2008) An Improved Car Detection using Street Layer Extraction. In: Perš J (ed) *Computer Vision Winter Workshop (SPRS)*

Palubinskas G, Kurz F, Reinartz P (2008) Detection of traffic congestion in optical remote sensing imagery. In: *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*

Palubinskas G, Runge H (2007) Radar Signatures of a Passenger Car. *IEEE Geoscience and Remote Sensing Letters*, 4 (4): 644 – 648.

Papageorgiou C, Poggio T (2000) A Trainable System for Object Detection. *International Journal of Computer Vision*, 38 (1): 15–33.

Pel AJ, Bliemer MCJ, Hoogendoorn SP (2012) A review on travel behaviour modelling in dynamic traffic simulation models for evacuations. *Transportation*, 39 (1): 97–123.

Pelapur R, Bunyak F, Palaniappan K, Seetharaman G (2013) Vehicle detection and orientation estimation using the radon transform. In: Pellechia MF, Sorensen RJ, Palaniappan K (eds) *Geospatial InfoFusion III*, SPIE 8747.

Perrotton X, Sturzel M, Roux M (2009) Mining families of features for efficient object detection. In: *IEEE International Conference on Image Processing (ICIP)*: 857–860.

Perrotton X, Sturzel M, Roux M (2010) Implicit hierarchical boosting for multi-view object detection. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*

Polikar R (2006) Ensemble Based Systems in Decision Making. *IEEE Circuits and Systems Magazine*, 6 (3): 21–45.

Prewitt J (1970) *Object Enhancement and Extraction*. Academic Press.

Pucher J, ren Peng Z, Mittal N, Zhu Y, Korattyswaroopam N (2007) Urban Transport Trends and Policies in China and India: Impacts of Rapid Economic Growth. *Transport Reviews: A Transnational Transdisciplinary Journal*, 27 (4): 379–410.

Rehrmann V, Birkhoff M (1995) Echtzeitfähige Objektverfolgung in Farbbildern. In: *Tagungsband 1. Workshop Farbbildverarbeitung*: 36–39.

Reilly V, Idrees H, Shah M (2010) Detection and Tracking of Large Number of Targets in Wide Area Surveillance. In: *European Conference on Computer Vision (ECCV)*

Reinartz P, Lachaise M, Schmeer E, Krauss T, Runge H (2006) Traffic monitoring with serial images from airborne cameras. *ISPRS Journal of Photogrammetry and Remote Sensing*, 61 (3-4): 149–158. Theme Issue: Airborne and Spaceborne Traffic Monitoring.

Ren X, Ramanan D (2013) Histograms of Sparse Codes for Object Detection. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*

Robinson G (1977) Edge Detection by Compass Gradient Masks. *Computer Graphics and Image Processing*, 6: 492–501.

Rodehorst V, Koschan A (2006) Comparison and evaluation of feature point detectors. In: Gründig L, Altan M (eds) *Proc. of the 5th Int. Symposium - Turkish-German Joint Geodetic Days*: 8.

Rodriguez E, Morris CS, Belz JE (2006) A Global Assessment of the SRTM Performance. *Photogrammetric Engineering and Remote Sensing*, 72 (3): 249–260.

Roth PM, Sternig S, Grabner H, Bischof H (2009) Classifier grids for robust adaptive object detection. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*: 2727–2734.

Salehi B, Zhang Y, Zhong M (2012) Automatic Moving Vehicles Information Extraction From Single-Pass WorldView-2 Imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS)*, 5 (1): 135–145.

Salem M, Meffert B (2007) A Comparison between 2D- and 3D-Wavelet based Segmentation for Traffic Monitoring Systems. In: *International Conference on Intelligent Computing and Information Systems (ICICIS)*: 329–334.

Schapire R (1990) Strength of Weak Learnability. *Machine Learning*, 5: 197–227.

Schapire R, Freund Y, Bartlett P, Lee W (1998) Boosting the Margin: A New Explanation for the Effectiveness of Voting Methods. *The Annals of Statistics*, 26 (5): 1651–1686.

Scharr H (2000) *Optimal operators in digital image processing.* PhD thesis, Ruprecht-Karls-Universität, Heidelberg, Heidelberg.

Schneiderman H, Kanade T (2000) A statistical method for 3D object detection applied to faces and cars. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1: 746–751.

Schrank D, Lomax T, Eisele B (2011) *2011 Urban Mobility Report.* Texas Transportation Institute, Technical report.

Sezgin M, Sankur B (2004) Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 13 (1): 146–165.

Sharma G, Merry C, Goel P, McCord M (2006) Vehicle detection in 1-m resolution satellite and airborne imagery. *International Journal of Remote Sensing*, 27 (3-4): 779–797.

Shillman R, Schatz D (2011) Video traffic monitoring and signaling apparatus. US Patent US8018352B2. Cognex Corporation.

Shimoni M, Tolt G, Perneel C, Ahlberg J (2011) Detection of Vehicles in Shadow Areas using Combined Hyperspectral and Lidar Data. In: *IEEE International Geoscience and Remote Sensing Symposium (IGARRS)*: 4427–4430.

Sobel I (1970) *Camera Models and Machine Perception.* PhD thesis, Stanford University.

Stantchev D, Whiteing T (2010) *Environmental Aspects - Thematic Research Summary.* European Commission DG Energy and Transport - Transport Research Knowledge Centre, Technical report.

Stilla U (1995) Map-aided structural analysis of aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 50 (4): 3–10.

Stilla U, Michaelsen E (2002) Estimating vehicle activity using thermal image sequences and maps. In: *Symposium on geospatial theory, processing and applications*, 34 (4).

Stilla U, Michaelsen E, Soergel U, Hinz S, Ender J (2004) Airborne Monitoring of Vehicle Activity in Urban Areas. In: Altan M (ed) *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*: Istanbul, Turkey, 34 (Part B3): 973–979.

Stilla U, Rottensteiner F, Hinz S, eds (2005) *Object Extraction for 3D City Models, Road Databases and Traffic Monitoring - Concepts, Algorithms, and Evaluation (CMRT)*, volume XXXVI (3/W24), Vienna.

Stilla U, Rottensteiner F, Paparoditis N, eds (2009) *Object Extraction for 3D City Models, Road Databases and Traffic Monitoring - Concepts, Algorithms and Evaluation (CMRT)*, volume XXXVIII (3/W4), Paris.

Sun M, Bao SY, Savarese S (2012) Object Detection using Geometrical Context Feedback. *International Journal of Computer Vision*, 100 (2): 154–169.

Sun Z, Bebis G, Miller R (2006) On-Road Vehicle Detection: A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 28 (5): 694–711.

Taylor R (2010) *2010 Urban Congestion Trends - Enhancing System Reliability with Operations*. U.S. Department of Transportation - Federal Highway Administration, Technical report. (accessed 17.10.2011).

Tele Atlas (1984) Parent is TomTom. http://navigation.teleatlas.com/. (accessed 19/11/2012).

TomTom (2009) White paper - How TomTom's HD Traffic and IQ Routes data provides the very best routing - Travel Time Measurements using GSM and GPS Probe Data. http://www.tomtom.com/lib/doc/download/HDT_White_Paper.pdf (accessed 21/08/2013).

Torralba A (2003) Contextual Priming for Object Detection. *International Journal of Computer Vision*, 53 (2): 169–191.

Torralba A, Murphy K, Freeman W (2005) Contextual models for object detection using boosted random fields. In: *Advances in Neural Information Processing Systems (NIPS)*: 1401–1408.

Torrent A, Llado X, Freixenet J, Torralba A (2011) Simultaneous detection and segmentation for generic objects. In: *IEEE International Conference on Image Processing (ICIP)*

Toth C (2009) The State-of-the Art in Airborne Data Collection Systems – Focused on LiDAR. *Photogrammetrische Woche 2009*, 18: 147–161.

Triggs B, McLauchlan PF, Hartley RI, Fitzgibbon AW (2000) *Bundle Adjustment – A Modern Synthesis*. Vision Algorithms '99, LNCS 1883. Springer-Verlag, Berlin Heidelberg.

Tsai LW, Hsieh JW, Fan KC (2005) Vehicle detection using normalized color and edge map. In: *IEEE International Conference on Image Processing (ICIP)*

Tuermer S, Kurz F, Reinartz P, Stilla U (2012) Airborne traffic monitoring supported by fast calculated digital surface models. In: *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*

Tuermer S, Kurz F, Reinartz P, Stilla U (2013) Airborne vehicle detection in dense urban areas using HoG features and disparity maps. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS)*, 6 (6): 2327–2337.

Tuermer S, Leitloff J, Reinartz P, Stilla U (2011a) Motion Component supported Boosted Classifier for Car Detection in Aerial Imagery. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science*, 38, Part 3 / W22.

Tuermer S, Leitloff J, Reinartz P, Stilla U (2011b) Vehicle Detection in Aerial Images using Boosted Classifier with Motion Mask. In: *Joint Urban Remote Sensing Event (JURSE)* (on CD).

U.S. Department of Transportation (2005) *Highway Statistics 2005*. U.S. Department of Transportation - Federal Highway Administration, Office of Highway Policy Information, Technical report.

U.S. Department of Transportation (2008) *Our Nation's Highways: 2008*. U.S. Department of Transportation - Federal Highway Administration (FHWA), Technical Report FHWA-PL-08021.

USGS (2000) United States Geological Survey. http://dds.cr.usgs.gov/srtm/. (accessed 19/11/2012).

Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1: I–511–I–518.

von Schönermark M (2010) Status Report on the Evaluation of the Radiometric Properties of Digital Photogrammetric Airborne Cameras. *Photogrammetrie - Fernerkundung - Geoinformation (PFG)*, 2010 (2): 131–139.

Voss F, Grüber B (2003) Verkehrslageerfassung aus der Luft, Verfahren zur automatisierten Auswertung von Thermal-Infrarot-Luftbildern. *Straßenverkehrstechnik*, 2: 75–82.

Wang CCR, Lien JJ (2008) Automatic Vehicle Detection Using Local Features: A Statistical Approach. *IEEE Transactions on Intelligent Transportation Systems*, 9 (1): 83–96.

Wang S (2011) Vehicle Detection on Aerial Images by Extracting Corner Features for Rotational Invariant Shape Matching. In: *IEEE 11th International Conference on Computer and Information Technology*: 171–175.

Wang X, Bai X, Liu W, Latecki L (2011) Feature context for image classification and object detection. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*: 961–968.

Winder A, Brackstone M, Debbah F (2010) *Transport Management*. European Commission DG Energy and Transport - Transport Research Knowledge Centre, Technical report.

Xiao J, Cheng H, Sawhney H, Han F (2010) Vehicle Detection and Tracking in Wide Field-of-View Aerial Video. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*: 679–684.

Xiao L, Chunping H, Chaoyun M, Baohua M (2011) Empirical study on variable lanes design of Chaoyang North Street in Beijing. In: *Chinese Control Conference (CCC)*: 5527–5531.

Yang J, Yu K, Gong Y, Huang T (2009) Linear spatial pyramid matching using sparse coding for image classification. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*

Yao W (2010) *Extraction and velocity estimation of vehicles in urban areas from airborne laser-scanning data*. PhD thesis, Technische Universität München (TUM).

Yao W, Hinz S, Stilla U (2009) Automatic estimation of vehicle activity from airborne thermal infrared video of urban areas by trajectory classification. *Photogrammetrie - Fernerkundung - Geoinformation (PFG)*, 2009 (5): 393–406.

Yao W, Hinz S, Stilla U (2011) Extraction and motion estimation of vehicles in single-pass airborne LiDARdata towards urban traffic analysis. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66 (3): 260–271.

Yao W, Stilla U (2011) Comparison of two methods for vehicle extraction from airborne LiDAR data toward motion analysis. *IEEE Geoscience and Remote Sensing Letters*, 8 (4): 607–611.

Yao W, Zhang M, Hinz S, Stilla U (2012) Airborne traffic monitoring in large areas using LiDAR data. *International Journal of Remote Sensing*, 33 (12): 3930–3945.

Zabih R, Woodfill J (1994) Non-parametric local transforms for computing visual correspondence. In: *European Conference on Computer Vision (ECCV)*: 151–158.

Zehnder P (2009) *Efficient Multi-Class Object Detection*. PhD thesis, ETH Zürich.

Zhao T, Nevatia R (2003) Car detection in low resolution aerial image. *Image and Vision Computing*, 21 (8): 693–703.

Zhou H, Jalayer M, Gong J, Hu S, Grinter M (2013) *Investigation Of Methods And Approaches For Collecting And Recording Highway Inventory Data*. Southern Illinois University Edwardsville, Research Report FHWA-ICT-13-022.

Zhou J, Gao D, Zhang D (2007) Moving Vehicle Detection for Automatic Traffic Monitoring. *IEEE Transactions on Vehicular Technology*, 56 (1): 51–59.

Zhu K, d'Angelo P, Butenuth M (2011) A Performance Study on Different Stereo Matching Costs Using Airborne Image Sequences and Satellite Images. In: *Photogrammetric Image Analysis (PIA) - Lecture Notes in Computer Science*, 6952: 159–170.

Zielstra D, Zipf A (2010) A Comparative Study of Proprietary Geodata and Volunteered Geographic Information for Germany. In: *International Conference on Geographic Information Science (AGILE)*

Acknowledgements

First of all, I would like to thank Prof. Uwe Stilla, who encouraged me at the beginning of this project. He supported the work throughout with interest and gave me the full benefit of his experience. Moreover, I would like to express my gratitude to Prof. Peter Reinartz who suggested the topic of vehicle detection and the position in the VABENE project. His insights and suggestions further improving the work were invaluable. In addition, I would like to thank Prof. Ralf Reulke and Prof. Urs Hugentobler.

Furthermore, I would like to thank Dr. Jens Leitloff who supervised me with great patience and introduced me to the field of machine learning. I would also like to thank Dr. Franz Kurz for sharing his knowledge of stereo vision computing and optical sensors. He also gave me very helpful criticism and feedback.

Last but not least, I would like to thank the whole department of Photogrammetry and Image Analysis at the German Aerospace Center (DLR) for the unstinting good humor and productive work environment. Special thanks goes to Peter Schwind for his valuable technical advice and wide-ranging discussions. I would also like to thank Rolf Stätter for proof reading.