Computer Vision for Autonomous Vehicles: Problems, Datasets and State-of-the-Art

Joel Janai[a,*], Fatma Güney[a,*], Aseem Behl[a,*], Andreas Geiger[a,b]

[a]Autonomous Vision Group, Max Planck Institute for Intelligent Systems, Spemannstr. 41, D-72076 Tübingen, Germany
[b]Computer Vision and Geometry Group, ETH Zürich, Universitätstrasse 6, CH-8092 Zürich, Switzerland

Abstract

Recent years have witnessed amazing progress in AI related fields such as computer vision, machine learning and autonomous vehicles. As with any rapidly growing field, however, it becomes increasingly difficult to stay up-to-date or enter the field as a beginner. While several topic specific survey papers have been written, to date no general survey on problems, datasets and methods in computer vision for autonomous vehicles exists. This paper attempts to narrow this gap by providing a state-of-the-art survey on this topic. Our survey includes both the historically most relevant literature as well as the current state-of-the-art on several specific topics, including recognition, reconstruction, motion estimation, tracking, scene understanding and end-to-end learning. Towards this goal, we first provide a taxonomy to classify each approach and then analyze the performance of the state-of-the-art on several challenging benchmarking datasets including KITTI, ISPRS, MOT and Cityscapes. Besides, we discuss open problems and current research challenges. To ease accessibility and accommodate missing references, we will also provide an interactive platform which allows to navigate topics and methods, and provides additional information and project links for each paper.

Keywords: Computer Vision, Autonomous Vehicles, Autonomous Vision

Since the first successful demonstrations in the 1980s (Dickmanns & Mysliwetz (1992); Dickmanns & Graefe (1988); Thorpe et al. (1988)), great progress has been made in the field of autonomous vehicles. Despite these advances, however, it is safe to believe that fully autonomous navigation in arbitrarily complex environments is still decades away. The reason for this is two-fold: First, autonomous systems which operate in complex dynamic environments require artificial intelligence which generalizes to unpredictable situations and reasons in a timely manner. Second, informed decisions require accurate perception, yet most of the existing computer vision systems produce errors at a rate which is not acceptable for autonomous navigation.

In this paper, we focus on the second aspect which we call *autonomous vision* and investigate the performance of current perception systems for autonomous vehicles. Towards this goal, we first provide a taxonomy of problems and classify existing datasets and techniques using this taxonomy, describing the pros and cons of each method. Second, we analyze the current state-of-the-art performance on several popular publicly available benchmarking datasets. In particular, we provide a novel in-depth qualitative analysis of the KITTI benchmark which shows the easiest and most difficult examples based on the methods submitted to the evaluation server. Based on this analysis, we discuss open research problems and challenges. To ease navigation, we also provide an interactive online tool which visualizes our taxonomy using a graph and provides additional information and links to project pages in an easily accessible

manner[1]. We hope that our survey will become a useful tool for researchers in the field of autonomous vision and lowers the entry barrier for beginners by providing an exhaustive overview over the field.

There exist several other related surveys. Winner et al. (2015) explains in detail systems for active safety and driver assistance, considering both their structure and their function. Their focus is to cover all aspects of driver assistance systems and the chapter about machine vision covers only the most basic concepts of the autonomous vision problem. Klette (2015) provide an overview over vision-based driver assistance systems. They describe most aspects of the perception problem at a high level, but do not provide an in-depth review of the state-of-the-art in each task as we pursue in this paper. Complementary to our survey, Zhu et al. (2017) provide an overview of environment perception for intelligent vehicles, focusing on lane detection, traffic sign/light recognition as well as vehicle tracking. In contrast, our goal is to bridge the gap between the robotics, intelligent vehicles, photogrammetry and computer vision communities by providing an extensive overview and comparison which includes works from all fields.

1. History of Autonomous Driving

1.1. Autonomous Driving Projects

Many governmental institutions worldwide started various projects to explore intelligent transportation systems (ITS). The

---

*Joint first authors with equal contribution.

[1]http://www.cvlibs.net/projects/autonomous_vision_survey

PROMETHEUS project started 1986 in Europe and involved more than 13 vehicle manufacturers, several research units from governments and universities of 19 European countries. One of the first projects in the United States was Navlab Thorpe et al. (1988) by the Carnegie Mellon University which achieved a major milestone in 1995, by completing the first autonomous drive[2] from Pittsburgh, PA and Sand Diego, CA. After many initiatives were launched by universities, research centers and automobile companies, the U.S. government established the National Automated Highway System Consortium (NAHSC) in 1995. Similar to the U.S., Japan established the Advanced Cruise-Assist Highway System Research Association in 1996 among many automobile industries and research centers to foster research on automatic vehicle guidance. Bertozzi et al. (2000) survey many approaches to the challenging task of autonomous road following developed during these projects. They concluded that sufficient computing power is become increasingly available, but difficulties like reflections, wet road, direct sunshine, tunnels and shadows still make data interpretation challenging. Thus, they suggested the enhancement of sensor capabilities. They also pointed out that the legal aspects related to the responsibility and impact of automatic driving on human passengers need to be considered carefully. In summary, the automation will likely be restricted to special infrastructures and will be extended gradually.

Motivated by the success of the PROMETHEUS projects to drive autonomously on highways, Franke et al. (1998) describe a real-time vision system for autonomous driving in complex urban traffic situations. While highway scenarios have been studied intensively, urban scenes have not been addressed before. Their system included depth-based obstacle detection and tracking from stereo as well as a framework for monocular detection and recognition of relevant objects such as traffic signs.

The fusion of several perception systems developed by VisLab[3] have led to several prototype vehicles including ARGO Broggi et al. (1999), TerraMax Braid et al. (2006), and BRAiVE Grisleri & Fedriga (2010). BRAiVE is the latest vehicle prototype which is now integrating all systems that VisLab has developed so far. Bertozzi et al. (2011) demonstrated the robustness of their system at the VisLab Intercontinental Autonomous Challenge, a semi-autonomous drive from Italy to China. The onboard system allows to detect obstacles, lane marking, ditches, berms and identify the presence and position of a preceding vehicle. The information produced by the sensing suite is used to perform different tasks such as leader-following and stop & go.

The PROUD project Broggi et al. (2015) slightly modified the BRAiVE prototype Grisleri & Fedriga (2010) to drive in urban roads and freeways open to regular traffic in Parma. Towards this goal they enrich an openly licensed map with information about the maneuver to be managed (e.g. pedestrian crossing, traffic light, . . .). The vehicle was able to handle complex situations such as roundabouts, intersections, priority roads, stops, tunnels, crosswalks, traffic lights, highways, and urban roads without any human intervention.

The V-Charge project Furgale et al. (2013) presents an electric automated car outfitted with close-to-market sensors. A fully operational system is proposed including vision-only localization, mapping, navigation and control. The project supported many works on different problems such as calibration Heng et al. (2013, 2015), stereo Häne et al. (2014), reconstruction Haene et al. (2012, 2013, 2014), SLAM Grimmett et al. (2015) and free space detection Häne et al. (2015). In addition to these research objectives, the project keeps a strong focus on deploying and evaluating the system in realistic environments.

Google started their self-driving car project in 2009 and completed over 1,498,000 miles autonomously until March 2016[4] in Mountain View, CA, Austin, TX and Kirkland, WA. Different sensors (i.a. cameras, radars, LiDAR, wheel encoder, GPS) allow to detect pedestrians, cyclists, vehicles, road work and more in all directions. According to their accident reports, Google's self-driving cars were involved only in 14 collisions while 13 times were caused by others. In 2016, the project was split off to Waymo[5], an independent self-driving technology company.

Tesla Autopilot[6] is an advanced driver assistant system developed by Tesla which was first rolled out in 2015 with version 7 of their software. The automation level of the system allows full automation but requires the full attention of the driver to take control if necessary. From October 2016, all vehicles produced by Tesla were equipped with eight cameras, twelve ultrasonic sensors and a forward-facing radar to enable full self-driving capability.

Long Distance Test Demonstrations: In 1995 the team within the PROMETHEUS project Dickmanns et al. (1990); Franke et al. (1994); Dickmanns et al. (1994) performed the first autonomous long-distance drive from Munich, Germany, to Odense, Denmark, at velocities up to 175 km/h with about 95% autonomous driving. Similarly, in the U.S. Pomerleau & Jochem (1996) drove from Washington DC to San Diego in the 'No hands across America' tour with 98% automated steering yet manual longitudinal control.

In 2014, Ziegler et al. (2014) demonstrated a 103 km ride from Mannheim to Pforzheim Germany, known as Bertha Benz memorial route, in nearly fully autonomous manner. They present an autonomous vehicle equipped with close-to-production sensor hardware. Object detection and free-space analysis is performed with radar and stereo vision. Monocular vision is used for traffic light detection and object classification. Two complementary vision algorithms, point feature based and lane marking based, allow precise localization relative to manually annotated digital road maps. They concluded that even thought the drive was successfully completed the overall behavior is far inferior to the performance level of an attentive human driver.

Recently, Bojarski et al. (2016) drove autonomously 98% of the time from Holmdel to Atlantic Highlands in Monmouth

---

[2] https://www.cmu.edu/news/stories/archives/2015/july/look-ma-no-hands.html
[3] http://www.vislab.it

[4] https://static.googleusercontent.com/media/www.google.com/lt//selfdrivingcar/files/reports/report-0316.pdf
[5] https://www.waymo.com
[6] https://www.tesla.com/autopilot

County NJ as well as 10 miles on the Garden State Parkway without intervention. Towards this goal, a convolutional neural network which predicts vehicle control directly from images is used in the NVIDIA DRIVE[TM] PX self-driving car. The system is discussed in greater detail in Section 11.

While all aforementioned performed impressively, the general assumption of precisely annotated road maps as well as pre-recorded maps for localization demonstrates that autonomous systems are still far from human capabilities. Most importantly, robust perception from visual information but also general artificial intelligence are required to reach human level reliability and react safely even in complex innercity situations.

1.2. Autonomous Driving Competitions

The European Land Robot Trial (ELROB)[7] is a demonstration and competition of unmanned systems in realistic scenarios and terrains, focusing mainly on military aspects such as reconnaissance and surveillance, autonomous navigation and convoy transport. In contrast to autonomous driving challenges, ELROB scenarios typically include navigation in rough terrain.

The first autonomous driving competition focusing on road scenes (though primarily dirt roads) has been initiated by the American Defense Advanced Research Projects Agency (DARPA) in 2004. The DARPA Grand Challenge 2004 offered a prize money of $1 million for the team first finishing a 150 mile route which crossed the border from California to Nevada. However, none of the robot vehicles completed the route. One year later, in 2005, DARPA announced a second edition of its challenge with 5 vehicles successfully completing the route (Buehler et al. (2007)). The third competition of the DARPA Grand Challenge, known as the Urban Challenge (Buehler et al. (2009)), took place on November 3, 2007 at the site of the George Air Force Base in California. The challenge involved a 96 km urban area course where traffic regulations had to be obeyed while negotiating with other vehicles and merging into traffic.

The Grand Cooperative Driving Challenge (GCDC[8], see also Geiger et al. (2012a)), a competition focusing on autonomous cooperative driving behavior was held in Helmond, Netherlands in 2011 for the first time and in 2016 for a second edition. During the competition, teams had to negotiate convoys, join convoys and lead convoys. The winner was selected based on a system that assigned points to randomly mixed teams.

2. Datasets & Benchmarks

Datasets have played a key role in the progress of many research fields by providing problem specific examples with ground truth. They allow quantitative evaluation of approaches providing key insights about their capacities and limitations. In particular, several of these datasets Geiger et al. (2012b); Scharstein & Szeliski (2002); Baker et al. (2011); Everingham et al. (2010); Cordts et al. (2016) also provide online evaluation servers which allow for a fair comparison on held-out test sets and provide researchers in the field an up-to-date overview over the state-of-the-art. This way, current progress and remaining challenges can be easily identified by the research community. In the context of autonomous vehicles, the KITTI dataset Geiger et al. (2012b) and the Cityscapes dataset Cordts et al. (2016) have introduced challenging benchmarks for reconstruction, motion estimation and recognition tasks, and contributed to closing the gap between laboratory settings and challenging real-world situations. Only a few years ago, datasets with a few hundred annotated examples were considered sufficient for many problems. The introduction of datasets with many hundred to thousands of labeled examples, however, has led to spectacular breakthroughs in many computer vision disciplines by training high-capacity deep models in a supervised fashion. However, collecting a large amount of annotated data is not an easy endeavor, in particular for tasks such as optical flow or semantic segmentation. This initiated a collective effort to produce that kind of data in several areas by searching for ways to automate the process as much as possible such as through semi-supervised learning or synthesization.

2.1. Real-World Datasets

While several algorithmic aspects can be inspected using synthetic data, real-world datasets are necessary to guarantee performance of algorithms in real situations. For example, algorithms employed in practice need to handle complex objects and environments while facing challenging environmental conditions such as direct lighting, reflections from specular surfaces, fog or rain. The acquisition of ground truth is often labor intensive because very often this kind of information cannot be directly obtained with a sensor but requires tedious manual annotation. For example, (Scharstein & Szeliski (2002),Baker et al. (2011)) acquire dense pixel-level annotations in a controlled lab environment whereas Geiger et al. (2012b); Kondermann et al. (2016) provide sparse pixel-level annotations of real street scenes using a LiDAR laser scanner.

Recently, crowdsourcing with Amazon's Mechanical Turk[9] have become very popular to create annotations for large scale datasets, e.g., Deng et al. (2009); Lin et al. (2014); Leal-Taixé et al. (2015); Milan et al. (2016). However, the annotation quality obtained via Mechanical Turk is often not sufficient to be considered as reference and significant efforts in post-processing and cleaning-up the obtained labels is typically required. In the following, we will first discuss the most popular computer vision datasets and benchmarks addressing tasks relevant to autonomous vision. Thereafter, we will focus on datasets particularly dedicated to autonomous vehicle applications.

Stereo and 3D Reconstruction: The Middlebury stereo benchmark[10] introduced by Scharstein & Szeliski (2002) provides several multi-frame stereo data sets for comparing the performance of stereo matching algorithms. Pixel-level ground truth is obtained by hand labeling and reconstructing planar components in piecewise planar scenes. Scharstein & Szeliski (2002)
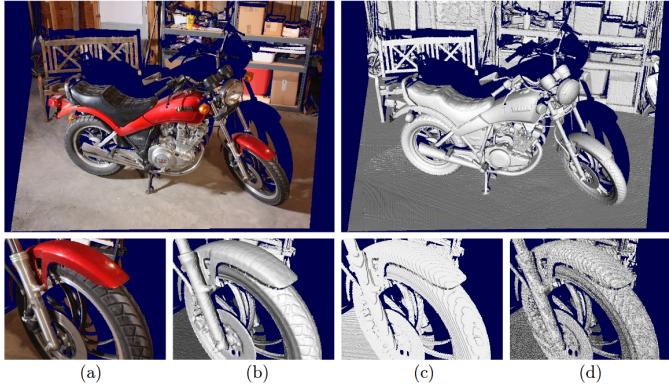

Figure 1: The structured light system of Scharstein et al. (2014) provides highly accurate depth ground truth, visualized in color and shadings (top). A close-up view is provided in (a),(b), rounded disparities are shown in (c) and the surface obtained using a baseline method in (d). Adapted from Scharstein et al. (2014).

further provide a taxonomy of stereo algorithms that allows the comparison of design decisions and a test bed for quantitative evaluation. Approaches submitted to their benchmark website are evaluated using the root mean squared error and the percentage of bad pixels between the estimated and ground truth disparity maps.

Scharstein & Szeliski (2003) and Scharstein et al. (2014) introduced novel datasets to the Middlebury benchmark comprising more complex scenes and including ordinary objects like chairs, tables and plants. In both works a structured lighting system was used to create ground truth. For the latest version Middlebury v3, Scharstein et al. (2014) generate highly accurate ground truth for high-resolution stereo images with a novel technique for 2D subpixel correspondence search and self-calibration of cameras as well as projectors. This new version achieves significantly higher disparity and rectification accuracy than those of existing datasets and allows a more precise evaluation. An example depth map from the dataset is illustrated in Figure 1.

The Middlebury multi-view stereo (MVS) benchmark[11] by Seitz et al. (2006) is a calibrated multi-view image dataset with registered ground truth 3D models for the comparison of MVS approaches. The benchmark played a key role in the advances of MVS approaches but is relatively small in size with only two scenes. In contrast, the TUD MVS dataset[12] by Jensen et al. (2014) provides 124 different scenes that were also recorded in controlled laboratory environment. Reference data is obtained by combining structured light scans from each camera position and the resulting scans are very dense, each containing 13.4 million points on average. For 44 scenes the full 360 degree model was obtained by rotation and scanning four times with 90 degree intervals. In contrast to the datasets so far, Schöps et al. (2017) provide scenes that are not carefully staged in a controlled laboratory environment and thus represent real world challenges. Schöps et al. (2017) recorded high-resolution DSLR imagery as well as synchronized low-resolution stereo

videos in a variety of indoor and outdoor scenes. A high-precision laser scanner allows to register all images with a robust method. The high-resolution images enable the evaluation of detailed 3D reconstruction while the low-resolution stereo images are provided to compare approaches for mobile devices.

Optical Flow: The Middlebury flow benchmark[13] by Baker et al. (2011) provides sequences with non-rigid motion, synthetic sequences and a subset of the Middlebury stereo benchmark sequences (static scenes) for the evaluation of optical flow methods. For all non-rigid sequences, ground truth flow is obtained by tracking hidden fluorescent textures sprayed onto the objects using a toothbrush. The dataset comprises eight different sequences with eight frames each. Ground truth is provided for one pair of frames per sequence.

Besides the limited size, real world challenges like complex structures, lighting variation and shadows are missing as the dataset necessitates laboratory conditions which allow for manipulating the light source between individual captures. In addition, it only comprises very small motions of up to twelve pixels which do not admit the investigation of challenges provided by fast motions. Compared to other datasets, however, the Middlebury dataset allows to evaluate sub-pixel precision since it provides very accurate and dense ground truth. Performance is measured using the angular error (AEE) and the absolute end point error (EPE) between the estimated flow and the ground truth.

Janai et al. (2017) present a novel optical flow dataset comprising of complex real world scenes in contrast to the laboratory setting in Middlebury. High-speed video cameras are used to create accurate reference data by tracking pixel through densely sampled space-time volumes. This method allows to acquire optical flow ground truth in challenging everyday scenes in an automatic fashion and to augment realistic effects such as motion blur to compare methods in varying conditions. Janai et al. (2017) provide 160 diverse real-world sequences of dynamic scenes with a significantly larger resolution (1280×1024 Pixels) than previous optical datasets and compare several state-of-the-art optical techniques on this data.

Object Recognition and Segmentation: The availability of large-scale, publicly available datasets such as ImageNet (Deng et al. (2009)), PASCAL VOC (Everingham et al. (2010)), Microsoft COCO (Lin et al. (2014)), Cityscapes (Cordts et al. (2016)) and TorontoCity (Wang et al. (2016)) have had a major impact on the success of deep learning in object classification, detection, and semantic segmentation tasks.

The PASCAL Visual Object Classes (VOC) challenge[14] by Everingham et al. (2010) is a benchmark for object classification, object detection, object segmentation and action recognition. It consists of challenging consumer photographs collected from Flickr with high quality annotations and contains large variability in pose, illumination and occlusion. Since its introduction, the VOC challenge has been very popular and was

---

[11]http://vision.middlebury.edu/mview/
[12]http://roboimagedata.compute.dtu.dk/?page_id=36

[13]http://vision.middlebury.edu/flow/
[14]http://host.robots.ox.ac.uk/pascal/VOC/

yearly updated and adapted to the needs of the community until the end of the program in 2012. Whereas the first challenge in 2005 had only 4 different classes, 20 different object classes were introduced in 2007. Over the years, the benchmark grew in size reaching a total of 11,530 images with 27,450 ROI annotated objects in 2012.

In 2014, Lin et al. (2014) introduced the Microsoft COCO dataset[15] for the object detection, instance segmentation and contextual reasoning. They provide images of complex everyday scenes containing common objects in their natural context. The dataset comprises 91 object classes, 2.5 million annotated instances and 328k images in total. Microsoft COCO is significantly larger in the number of instances per class than the PASCAL VOC object segmentation benchmark. All objects are annotated with per-instance segmentations in an extensive crowd worker effort. Similar to PASCAL VOC, the intersection-over-union metric is used for evaluation.

Tracking: Leal-Taixé et al. (2015); Milan et al. (2016) present the MOTChallenge[16] which addresses the lack of a centralized benchmark for multi object tracking. The benchmark contains 14 challenging video sequences in unconstrained environments filmed with static and moving cameras and subsumes many existing multi-object tracking benchmarks such as PETS (Ferryman & Shahrokni (2009)) and KITTI (Geiger et al. (2012b)). The annotations for three object classes are provided: moving or standing pedestrians, people that are not in an upright position and others. They use the two popular tracking measures, Multiple Object Tracking Accuracy (MOTA) and Multiple Object Tracking Precision (MOTP) introduced by Stiefelhagen et al. (2007) for the evaluation of the approaches. Detection ground truth provided by the authors allows to analyze the performance of tracking systems independent of a detection system. Methods using a detector and methods using the detection ground truth can be compared separately on their website.

Aerial Image Datasets: The ISPRS benchmark[17] (Rottensteiner et al. (2013, 2014)) provides data acquired by airborne sensors for urban object detection and 3D building reconstruction and segmentation. It consists of two datasets: Vaihingen and Downtown Toronto. The object classes considered in the object detection task are building, road, tree, ground, and car. The Vaihingen dataset provides three areas with various object classes and a large test site for road detection algorithms. The Downtown Toronto dataset covers an area of about 1.45 km^2 in the central area of Toronto, Canada. Similarly to Vaihingen, there are two smaller areas for object extraction and building reconstruction, as well as one large area for road detection. For each test area, aerial images with orientation parameters, digital surface model (DSM), orthophoto mosaic and airborne laser scans are provided. The quality of the approaches is assessed using several metrics for detection and reconstruction. In both cases completeness, correctness and quality is assessed on a per-area level and a per-object level.
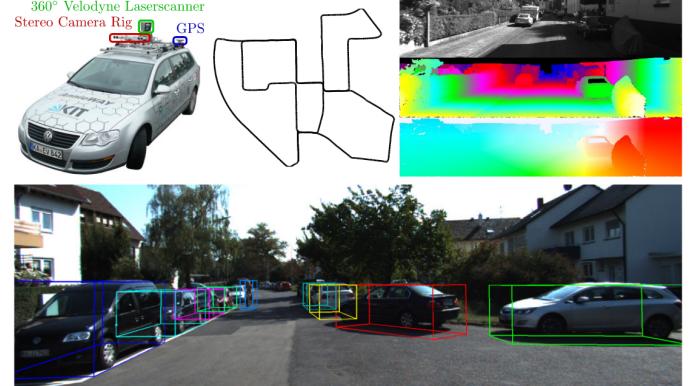

Figure 2: The recording platform with sensors (top left), trajectory (top center), disparity and optical flow (top right) and 3D object labels (bottom) from the KITTI benchmark proposed by Geiger et al. (2012b). Adapted from Geiger et al. (2012b).

Autonomous Driving: In 2012, Geiger et al. (2012b, 2013) have introduced the KITTI Vision Benchmark[18] for stereo, optical flow, visual odometry/SLAM and 3D object detection (Figure 2). The dataset has been captured from an autonomous driving platform and comprises six hours of recordings using high-resolution color and grayscale stereo cameras, a Velodyne 3D laser scanner and high-precision GPS/IMU inertial navigation system. The stereo and optical flow benchmarks derived from this dataset comprise 194 training and 195 test image pairs at a resolution of 1280×376 pixels and sparse ground truth obtained by projecting accumulated 3D laser point clouds onto the image. Due to the limitations of the rotating laser scanner used as reference sensor, the stereo and optical flow benchmark is restricted to static scenes with camera motion.

To provide ground truth motion fields for dynamic scenes, Menze & Geiger (2015) have annotated 400 dynamic scenes, fitting accurate 3D CAD models to all vehicles in motion in to order to obtain flow and stereo ground truth for these objects. The KITTI flow and stereo benchmarks use the percentage of erroneous (bad) pixels to assess the performance of the submitted methods. Additionally, Menze & Geiger (2015) combined the stereo and flow ground truth to form a novel 3D scene flow benchmark. For evaluating scene flow, they combine classical stereo and optical flow measures.

The visual odometry / SLAM challenge consists of 22 stereo sequences, with a total length of 39.2 km. The ground truth pose is obtained using GPS/IMU localization unit which was fed with RTK correction signals. The translational and rotational error averaged over a particular trajectory length is considered for evaluation.

For the KITTI object detection challenge, a special 3D labeling tool has been developed to annotate all 3D objects with 3D bounding boxes for 7481 training and 7518 test images. The benchmark for the object detection task was separated into a vehicle, pedestrian and cyclist detection tasks, allowing to focus the analysis on the most important problems in the context

---

[15]http://mscoco.org/
[16]https://motchallenge.net/
[17]http://www2.isprs.org/commissions/comm3/wg4/tests.html
[18]http://www.cvlibs.net/datasets/kitti/

of autonomous vehicles. Following PASCAL VOC Everingham et al. (2010), the intersection-over-union (IOU) metric is used for evaluation. For an additional evaluation, this metric has been extended to capture both 2D detection and 3D orientation estimation performance. A true 3D evaluation is planned to be released shortly.

The KITTI benchmark was extended by Fritsch et al. (2013) to the task of road/lane detection. In total, 600 diverse training and test images have been selected for manual annotation of road and lane areas. Mattyus et al. (2016) used aerial images to enhance the KITTI dataset with fine grained segmentation categories such as parking spots and sidewalk as well as the number and location of road lanes. The KITTI dataset has established itself as one of the standard benchmarks in all of the aforementioned tasks, in particular in the context of autonomous driving applications.

Complementary to other datasets, the HCI benchmark[19] proposed in Kondermann et al. (2016) specifically includes realistic, systematically varied radiometric and geometric challenges. Overall, a total of 28,504 stereo pairs with stereo and flow ground truth is provided. In contrast to previous datasets, ground truth uncertainties have been estimated for all static regions. The uncertainty estimate is derived from pixel-wise error distributions for each frame which are computed based on Monte Carlo sampling. Dynamic regions are manually masked out and annotated with approximate ground truth for 3,500 image pairs.

The major limitation of this dataset is that all sequences were recorded in a single street section, thus lacking diversity. On the other hand, this enabled better control over the content and environmental conditions. In contrast to the mobile laser scanning solution of KITTI, the static scene is scanned only once using a high-precision laser scanner in order to obtain a dense and highly accurate ground truth of all static parts. Besides the metrics used in KITTI and Middlebury, they use semantically meaningful performance metrics such as edge fattening and surface smoothness for evaluation Honauer et al. (2015). The HCI benchmark is rather new and not established yet but the controlled environment allows to simulate rarely occurring events such as accidents which are of great interest in the evaluation of autonomous driving systems.

The Caltech Pedestrian Detection Benchmark[20] proposed by Dollar et al. (2009) provides 250,000 frames of sequences recorded by a vehicle while driving through regular traffic in an urban environment. 350,000 bounding boxes and 2,300 unique pedestrians were annotated including temporal correspondence between bounding boxes and detailed occlusion labels. Methods are evaluated by plotting the miss rate against false positives and varying the threshold on detection confidence.

The Cityscapes Dataset[21] by Cordts et al. (2016) provides a benchmark and large-scale dataset for pixel-level and instance-level semantic labeling that captures the complexity of real-world urban scenes. It consists of a large, diverse set of stereo video sequences recorded in streets of different cities. High quality pixel-level annotations are provided for 5,000 images while 20,000 additional images have been annotated with coarse labels obtained using a novel crowd sourcing platform. For two semantic granularities, i.e., classes and categories, they report mean performance scores and evaluate the intersection-over-union metric at instance-level to assess how well individual instances are represented in the labeling.

The TorontoCity benchmark presented by Wang et al. (2016) covers the greater Toronto area with 712 km^2 of land, 8,439 km of road and around 400,000 buildings. The benchmark covers a large variety of tasks including building height estimation (reconstruction), road centerline and curb extraction, building instance segmentation, building contour extraction, semantic labeling and scene type classification. The dataset was captured from airplanes, drones, and cars driving around the city to provide different perspectives.

Long-Term Autonomy: Several datasets such as KITTI or Cityscapes focus on the development of algorithmic competences for autonomous driving but do not address challenges of long-term autonomy, as for examples environmental changes over time. To address this problem, a novel dataset for autonomous driving has been presented by Maddern et al. (2016). They collected images, LiDAR and GPS data while traversing 1,000 km in central Oxford in the UK during one year. This allowed them to capture large variations in scene appearance due to illumination, weather and seasonal changes, dynamic objects, and constructions. Such long-term datasets allow for in-depth investigation of problems that detain the realization of autonomous vehicles such as localization in different times of the year.

2.2. Synthetic Data

The generation of ground truth for real examples is very labor intensive and often not even possible at large scale when pixel-level annotations are required. On the other hand, pixel-level ground truth for large-scale synthetic datasets can be easily acquired. However, the creation of realistic virtual world is time consuming. The popularity of movies and video games have led to an industry creating very realistic 3D content which nourishes the hope to replace real data completely using synthetic datasets. Consequently, several synthetic datasets have been proposed, recently, but it remains an open question whether the realism and variety attained is sufficient to replace real world datasets. Besides, creating realistic virtual content is a time consuming and expensive process itself and the trade-off between real and synthetic (or augmented) data is not clear yet.

MPI Sintel: The MPI Sintel Flow benchmark[22] presented by Butler et al. (2012) takes advantage of the open source movie Sintel, a short animated film, to render scenes of varying complexity with optical flow ground truth. In total, Sintel comprises 1,628 frames. Different datasets obtained using different passes of the rendering pipeline vary in complexity shown in Figure 3.

---

[19]http://hci-benchmark.org
[20]http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/
[21]https://www.cityscapes-dataset.com/

[22]http://sintel.is.tue.mpg.de/

Figure 3: This figure was adapted from Butler et al. (2012) and shows the varying complexity of the Sintel benchmark obtained with different passes of the rendering pipeline: albedo, clean and final (from top to bottom).

The albedo pass has roughly piecewise constant colors without illumination effects while the clean pass introduces illumination of various kinds. The final pass adds atmospheric effects, blur, color correction and vignetting. In addition to the average endpoint error, the benchmark website provides different rankings of the methods based on speed, occlusion boundaries, and disocclusions.

Flying Chairs and Flying Things: The limited size of optical flow datasets hampered the training of deep high-capacity models. To train a convolutional neural network, Dosovitskiy et al. (2015) thus introduced a simple synthetic 2D dataset of flying chairs rendered on top of random background images from Flickr. As the limited realism and size of this dataset proved insufficient to learn highly accurate models, Mayer et al. (2016) presented another large-scale dataset consisting of three synthetic stereo video datasets: FlyingThings3D, Monkaa, Driving. FlyingThings3D provides everyday 3D objects flying along randomized 3D trajectories in a randomly created scene. Inspired by the KITTI dataset a driving dataset has been created which uses car models from the same pool as FlyingThings3D and additionally highly detailed tree and building models from 3D Warehouse. Monkaa is an animated short movie similar to Sintel used in the MPI Sintel benchmark.

Game Engines: Unfortunately, data from animated movies is very limited since the content is hard to change and such movies are rarely open source. In contrast, game engines allow for creating an infinite amount of data. One way to create virtual worlds using a game engine is presented by Gaidon et al. (2016) which introduces the Virtual KITTI dataset[23]. They present an efficient real-to-virtual world cloning method to create realistic proxy worlds. A cloned virtual world allows to vary conditions such as weather or illumination and to use different camera settings. This way, the proxy world can be used for virtual data augmentation to train deep networks. Virtual KITTI contains 35 photo-realistic synthetic videos with a total of 17,000 high resolution frames. They provide ground truth for object detection, tracking, scene and instance segmentation, depth and optical flow.

In concurrent work, Ros et al. (2016) created SYNTHIA[24], a synthetic collection of Imagery and Annotations of urban scenarios for semantic segmentation. They rendered a virtual city with the Unity Engine. The dataset consists of 13,400 randomly taken virtual images from the city and four video sequences with 200,000 frames in total. Pixel-level semantic annotations are provided for 13 classes.

Richter et al. (2016) have extracted pixel-accurate semantic label maps for images from the commercial video game Grand Theft Auto V. Towards this goal, they developed a wrapper which operates between the game and the graphics hardware to obtain pixel-accurate object signatures across time and instances. The wrapper allows them to produce dense semantic annotations for 25 thousand images synthesized by the photo-realistic open-world computer game with minimal human supervision. However, for legal reasons, the extracted 3D geometry can not be made publicly available. Similarly, Qiu & Yuille (2016) provide an open-source tool to create virtual worlds by accessing and modifying the internal data structure of Unreal Engine 4. They show how virtual worlds can be used to test deep learning algorithms by linking them with the deep learning framework Caffe Jia et al. (2014).

3. Cameras Models & Calibration

3.1. Calibration

Multiple sensors including odometry, range sensors, and different types of cameras such as perspective and fish-eye are widely used in automotive context. Calibration is the problem of estimating intrinsic and extrinsic parameters of these sensors to relate 2D image points to 3D world points and represent sensed information in a common coordinate system in case of multiple sensors. Fiducial markings on checkerboard patterns are the standard tool for calibration. Almost all systems use them either for initialization or for joint optimization to improve the intrinsics. Reprojection error which is the pixel distance between a projected point and a measured one, is used as a way of measuring accuracy quantitatively. Accuracy of calibration is a key issue in driver assistance applications requiring 3D reasoning, and consequently in the safety of autonomous vehicles. Besides accuracy, other desired qualities in a calibration system are speed, robustness to varying imaging conditions, full automation, minimum restrictions in terms of assumptions such as overlapping field of view or information required such as an initial guess of the parameters.

Modern systems are equipped with multiple sensors for different purposes. Geiger et al. (2012c) use a setup involving two cameras and a single range sensor such as Kinect or Velodyne laser scanner. They present two algorithms for camera-to-camera and camera-to-range calibration using a single image per sensor. They assume a common field of view for the sensors which is particularly useful for applications such as generating stereo or scene flow ground truth. Heng et al. (2013)

---

and Heng et al. (2015) tackle the automatic intrinsic and extrinsic calibration of a multi-camera rig system with four fish-eye cameras and odometry without assuming overlapping fields of view. Heng et al. (2015) propose an improved version of Heng et al. (2013). While Geiger et al. (2012c) require fiducial markings to re-calibrate the system before every run, they remove the requirement to modify infrastructure by using a map and natural features instead. They first build a map of the calibration area and then perform calibration by using this map and image-based geo-localization. In contrast to SLAM-based self-calibration methods, image-based localization removes the burden of exhaustive feature matching between different cameras and bundle adjustment.

3.2. Omnidirectional Cameras

A panoramic field of view is desirable in autonomous driving to gain maximum information about the surrounding area for safe navigation. An omnidirectional camera with a 360-degree field of view provides enhanced coverage by eliminating the need for more cameras or mechanically turnable cameras. There are different types of omnidirectional cameras with a visual field that covers a hemisphere or even approximately the entire sphere. Catadioptric cameras combine a standard camera with a shaped mirror, such as a parabolic, hyperbolic, or elliptical mirror while dioptric cameras use purely dioptric fisheye lenses. Polydioptric cameras use multiple cameras with overlapping field of view to provide a full spherical field of view.

One classification often used in the literature for omnidirectional cameras is based on the projection center: central and noncentral. In central cameras, the optical rays to the viewed objects intersect in a single point in 3D which is known as the single effective viewpoint property. This property allows the generation of geometrically correct perspective images from the images captured by omnidirectional cameras and consequently, application of epipolar geometry which holds for any central camera. Central catadioptric cameras are built by choosing the mirror shape and the distance between the camera and the mirror.

In contrast to pinhole cameras, calibration of omnidirectional cameras cannot be modeled by a linear projection due to very high distortion. The model should take into account the reflection of the mirror in the case of a catadioptric camera or the refraction caused by the lens in the case of a fisheye camera. Geyer & Daniilidis (2000) provide a unifying theory for all central catadioptric systems which is known as unified projection model in the literature and widely used by different calibration toolboxes (Mei & Rives (2007); Heng et al. (2013, 2015)). They prove that every projection, both standard perspective and catadioptric using a hyperbolic, parabolic, or elliptical mirror, can be modeled with projective mappings from the sphere to a plane where projection center is on a sphere diameter and the plane perpendicular to it. Scaramuzza & Martinelli (2006) propose modeling the imaging function by a Taylor series expansion whose degree and the coefficients are the parameters to be estimated. Polynomials of order three or four are able to model accurately all catadioptric cameras and many types of fisheye cameras. Mei & Rives (2007) improve upon unified projection

model of Geyer & Daniilidis (2000) to account for real-world errors by modeling distortions with well identified parameters.

As desirable as it is, the single viewpoint property is often violated in practice due to varifocal lenses and difficulty of precise alignment. However, non-central models as the alternative are computationally demanding, hence not suitable for real-time applications. Schönbein et al. (2014) extend a non-central approach in order to accurately obtain the viewing ray orientations, and then propose a fast central approximation with a mapping to match the obtained orientations. This kind of approach, tested on hypercatadioptric cameras, achieves a reprojection error lower than the central models (Geyer & Daniilidis (2000); Scaramuzza & Martinelli (2006); Mei & Rives (2007)) and comparable to non-central models while being much faster.

Applications: Omnidirectional cameras are more and more used in autonomous driving. For feature based applications such as navigation, motion estimation and mapping, large field of view enables extraction and matching of interesting points from all around the car. For instance, omnidirectional feature matches improve the rotation estimate significantly when doing visual odometry or simultaneous localization and mapping (SLAM). Scaramuzza & Siegwart (2008) estimate the ego-motion of the vehicle relative to the road from a single, central omnidirectional camera by using a homography based tracker for the ground plane and an appearance-based tracker for the rotation of the vehicle. 3D perception also benefits from the unified view offered by omnidirectional sensors, despite the limited effective resolution which leads to noisy reconstructions. Laser-based solutions as an alternative provide only sparse point clouds without color, are extremely expensive and suffer from rolling shutter effects. Schönbein & Geiger (2014) propose a method for 3D reconstruction through joint optimization of disparity estimates from two temporally and two spatially adjacent omnidirectional view in a unified omnidirectional space by using plane-based priors. Häne et al. (2014) extend the plane-sweeping stereo matching for fisheye cameras by incorporating unified projection model for fisheye cameras directly into the plane-sweeping stereo matching algorithm. This kind of approach allows producing dense depth maps directly from fisheye images in real time using GPUs and opens the way for dense 3D reconstruction with large a field of view in real time.

3.3. Event Cameras

Contrary to conventional frame-based imagers at constant frame rates, event-based sensors have very recently been introduced. They produce a stream of asynchronous events at microsecond resolution in case of a brightness change surpassing a pre-defined threshold (Dynamic Vision Sensor) as shown in Figure 4. An event contains the location, sign, and precise timestamp of the change. This kind of data is sparse in nature, thus reducing redundancy in transmission and processing. Another advantage is high temporal resolution, allowing the design of highly reactive systems. These properties, namely low latency and low bandwidth requirement make event-based sensors interesting for autonomous driving. However, standard computer-vision algorithms cannot be applied directly to the
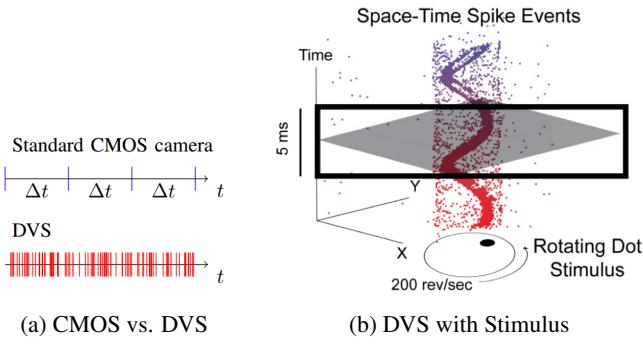
Figure 4: (a) A standard CMOS camera sends images at a fixed frame rate (blue) while a Dynamic Vision Sensor (DVS) sends spike events at the time they occur (red). Each event corresponds to a local, pixel-level change of brightness. (b) Visualization of the output of a DVS looking at a rotating dot. Colored dots mark individual events. Events that are not part of the spiral are caused by sensor noise. Adapted from Mueggler et al. (2015b).

output of event-based vision sensors which is fundamentally different from intensity images. Events occur at high frequency and each event doesn't carry enough information by itself. A straightforward solution is to generate intensity images by accumulating events over a fixed time interval, but this kind of event-to-frame conversion introduces some latency and obstructs the efficiency which comes with the high temporal resolution.

Instead, algorithms should ideally exploit the high rate at which events are generated. Consequently several methods have recently been introduced which exploit the high temporal resolution and the asynchronous nature of the sensor for different problems in autonomous vision. The design goal of such algorithms is that each incoming event can asynchronously change the estimated state, thus respecting the event-based nature of the sensor and allowing for perception and state estimation in highly dynamic scenarios. For trajectory estimation, Mueggler et al. (2015b) propose a continuous temporal model as a natural representation of the pose trajectory described by a smooth parametric model. Rebecq et al. (2016) propose an event-based 3D reconstruction algorithm to produce a parallel tracking and mapping pipeline that runs in real-time on the CPU. Event-based SLAM does not suffer from motion blur due to high speed motions and very high dynamic range scenes which can be challenging for standard camera approaches.

Lifetime Estimation: In addition to enabling novel solutions for existing problems where low latency and high frame rates are required, event-based sensors also give rise to new problems. One such problem is lifetime estimation of events by modeling the set of active events. An event is considered active as long as the brightness gradient causing the event is visible by the pixel. Explicit modeling of active events can be used to generate sharp gradient images at any point in time, or for clustering of events in tracking of multiple objects. For this task, Mueggler et al. (2015a) propose using event-based optical flow with optional regularization, independent of a temporal window.

4. Representations

A wide variety of representations at different levels of granularity is used in the computer vision literature. Variables or parameters can be associated directly with 2D pixels in an image or describe high-level primitives in 3D space. In pixel-based representation each pixel is a separate entity, for example a random variable in a graphical model. Pixels are amongst the most fine-grained representations, but are harder to relate to physical properties of our 3D world. Furthermore, pixel-based representations increase complexity of inference algorithms due to the large number of variables in high resolution images. As a consequence, many approaches model only local interactions between pixels which do not capture the structure of our world sufficiently well to overcome all ambiguities in the ill-posed inverse problems computer vision is trying to solve.

Superpixels: Consequently, compact representations based on grouping of pixels, i.e. superpixels, have gained popularity. Superpixel-based representations are obtained by a segmentation of the image into atomic regions which are ideally similar in color and texture, and respect image boundaries (Ren & Malik (2003); Achanta et al. (2012); Li & Chen (2015)). The implicit assumption each superpixel-based method makes is that certain properties of interest remain constant within a superpixel, e.g., the semantic class label or the slant of a surface. However, boundary adherence with respect to these properties is easily violated, especially for cluttered images when relying on standard segmentation algorithms which leverage color or intensity cues.

If available, depth information can be leveraged as valuable feature for accurate superpixel extraction (Badino et al. (2009); Yamaguchi et al. (2014)). Superpixels are used as building blocks for various tasks such as stereo and flow estimation (Yamaguchi et al. (2012, 2013, 2014); Güney & Geiger (2015); Bai et al. (2016)), scene flow (Menze & Geiger (2015); Menze et al. (2015b); Lv et al. (2016)), semantic segmentation (Xiao & Quan (2009); Wegner et al. (2013)), scene understanding (Ess et al. (2009b); Liu et al. (2014)) and 3D reconstruction (Schönbein et al. (2014)). In cases that include geometric reasoning such as stereo estimation, superpixels often represent 3D planar segments. When the goal is to represent real-world scenes with independent object motion as in scene flow or optical flow, superpixels can be generalized to rigidly moving segments (Vogel et al. (2015); Menze & Geiger (2015)), or semantic segments (Bai et al. (2016); Sevilla-Lara et al. (2016)).

Stixels: Stixels are presented as a medium level representation of 3D traffic scenes with the goal to bridge the gap between pixels and objects (Badino et al. (2009)). The so-called "Stixel World" representation originates from the observation that free space in front of the vehicle is mostly limited by vertical surfaces. Stixels are represented by a set of rectangular sticks standing vertically on the ground to approximate these surfaces. Assuming a constant width, each stixel is defined by its 3D position relative to the camera and its height. The main goal is to gain efficiency through a compact, complete, stable, and robust representation. In addition, Stixel representations

Figure 5: The multi-layer Stixel World representation of Pfeiffer & Franke (2011). The scene is segmented into planar segments termed "Stixels". In contrast to the Stixel World of Badino et al. (2009), objects are allowed to be located at multiple depths within a single image column. The color represents the distance to the obstacle with red being close and green far away. Adapted from Pfeiffer & Franke (2011).

provide an encoding of the free space and the obstacles in the scene.

Using depth maps from SGM Hirschmüller (2008) as input, Badino et al. (2009) use dynamic programming based on occupancy grids to compute free space (determining the Stixels' lower positions) and foreground/background segmentation on the disparity map (to compute the Stixels' height). Pfeiffer & Franke (2011) extend Badino et al. (2009) to a unified probabilistic scheme. They lift the constraint of Stixels to touch the ground and allow multiple stixels along an image column. This way objects can be located at multiple depths in a single image column (Figure 5).

Pfeiffer & Franke (2010) extend the Stixel world representation to dynamic scenes by tracking stixels using a 6D Kalman filter framework and optical flow as input. Erbs et al. (2012, 2013) propose a CRF framework for segmenting a traffic scene based on the Dynamic Stixel World representation. Günyel et al. (2012) show that motion estimation for stixels can be reduced to a 1D problem and can be solved efficiently via 2D dynamic programming by avoiding costly dense optical flow computation.

Levi et al. (2015) propose to use a CNN called StixelNet for learning to extract the foot point of each Stixel from the image. Cordts et al. (2014) propose to incorporate top-down object-level cues into bottom-up Stixel representation in a probabilistic approach. In order to achieve that, they leverage probability images derived from the output of three different object detectors, namely pedestrian, vehicle, and guard rail. Schneider et al. (2016) propose a semantic Stixel representation to jointly infer semantic and geometric layout of the scene from a dense disparity map and a pixel-level semantic scene labeling.

3D Primitives: The use of 3D geometric primitives is very common in 3D reconstruction, particularly when reconstructing urban areas. Atomic regions which are geometrically meaningful allow the shape of urban objects to be better preserved. In addition, simplified geometric assumptions can provide significant speedups as well as a compact model. In Cornelis et al. (2008), 3D city models are composed of ruled surfaces for both the facades and the roads. Duan & Lafarge (2016) use polygons with elevation estimate for 3D city modeling from pairs of satellite images. de Oliveira et al. (2016) update a list of large scale polygons over time for an incremental scene representation from 3D range measurements. Lafarge et al. (2010) use a

library of 3D blocks for reconstructing buildings with different roof forms. Lafarge & Mallet (2012); Lafarge et al. (2013) use 3D-primitives such as planes, cylinders, spheres or cones for describing regular structures of the scene. Dubé et al. (2016) segments point clouds into distinct elements for a loop-closure detection algorithm based on the matching of 3D segments.

5. Object Detection

Reliable detection of objects is a crucial requirement to realize autonomous driving. As the car is sharing the road with many traffic participants, particularly in urban areas, the awareness of other traffic participants or obstacles is necessary to avoid accidents that might be life threatening. The detection in urban areas is hard because of the wide variety of object appearances and occlusions caused by other objects or the object of interest itself. In addition, the resemblance of objects to each other or to the background and physical effects like cast shadows or reflections can make the distinction difficult.

Sensors: The object detection task can be addressed with a variety of different of sensors. Cameras are the cheapest and most commonly used type of sensors for the detection of objects. The visible spectrum (VS) is typically used for daytime detections whereas the infrared spectrum can be used for nighttime detection. Thermal infrared (TIR) cameras capture relative temperature which allows to distinguish warm objects like pedestrians from cold objects like vegetation or the road. Active sensors, that emit signals and observe their reflection, like laser scanners can provide range information which is helpful for detecting an object and localizing it in 3D. Depending on the weather conditions or material properties it can be problematic to rely on a single type of sensor alone. VS cameras and laser scanners are affected by reflective or transparent surfaces while hot objects (like engines) or warm temperatures can influence TIR cameras. The combination of information from different sensors via sensor fusion (Enzweiler & Gavrila (2011); Chen et al. (2016b); González et al. (2016)) allows for the robust integration of this complementary information.

Standard Pipeline: A traditional detection pipeline consists of the following steps: preprocessing, region of interest extraction (ROI), object classification and verification/refinement. In the preprocessing step tasks such as exposure and gain adjustment, as well as camera calibration and image rectification are usually performed. Some approaches leverage temporal information with a joint detection and tracking system. We give a detailed overview of the tracking problem in Section 9.

Regions of interest can be extracted using a sliding window approach which shifts a detector over over the image at different scales. As exhaustive search is very expensive, several heuristics have been proposed for reducing the search space. Typically, the number of evaluations is reduced by assuming a certain ratio, size and position of candidate bounding boxes. Apart from that, image features, stereo or optical flow can be leveraged for focusing the search on the relevant regions. Broggi et al. (2000) filter pedestrian candidates using morphological characteristics (size, ratio and shape) and vertical symmetry of
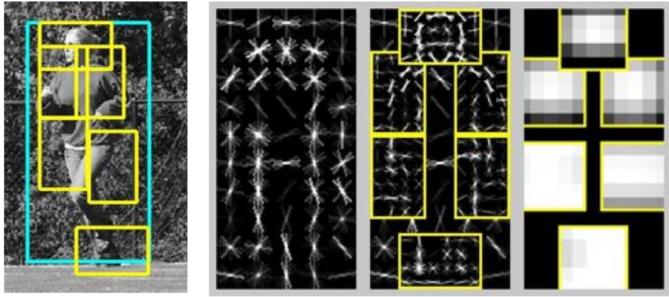
Figure 6: An example detection obtained with the Deformable Part Model proposed by Felzenszwalb et al. (2008). The DPM comprises a coarse as well as multiple high resolution models and a spatial constellation model for constraining the location of each part. Adapted from Felzenszwalb et al. (2008).

human shape. In addition, they exploit the distance information obtained from stereo vision in the ROI extraction and refinement steps of the algorithm. Selective Search (Uijlings et al. (2013)) is an alternative approach to generate regions of interest. They exploit segmentation for efficiently extracting approximate locations instead of performing an exhaustive search over the full image domain.

In their survey on pedestrian detection systems from monocular images, Dollar et al. (2011) present an extensive evaluation focusing on the evaluation of sliding window approaches. They claim that these approaches are most promising for low to medium resolution detection but found that detection with low resolution inputs and occlusions are still problematic for the considered approaches.

Classification: The classification of all candidates in an image using the sliding window approach can become quite costly due to the vast amount of image regions which need to be classified. Therefore, a fast decision is necessary which quickly discards candidates in the background region of the image. Viola et al. (2005) combine simple and efficient classifiers, learned using AdaBoost, in a cascade which allows to quickly discard false candidates while spending more time on promising regions. With the work of Dalal & Triggs (2005), linear Support Vector Machines (SVMs), that maximizes the margin of all samples from a linear decision boundary, in combination with Histogram of Orientation (HOG) features have become popular tools for classification. However, all previous methods rely on hand-crafted features that are difficult to design. With the renaissance of deep learning, convolutional neural networks have automated this task while significantly boosting performance. For example, Sermanet et al. (2013) introduced CNNs to the pedestrian detection problem using unsupervised convolutional sparse auto-encoders to pre-train features and end-to-end supervised training to train the classifier while fine-tuning the features. Today, all state of the art detection approaches are learned in an end-to-end fashion from large datasets as we will discuss in Section 5.1.

Part-based Approaches: Learning the appearance of articulated objects is difficult because all possible articulations need to be considered. The idea of part-based approaches is to split the complex appearance of non-rigidly moving objects like hu-

mans into simple parts and to represent any articulation using these parts. This provides greater flexibility and reduces the number of training examples required for learning the object appearance. The Deformable Part Model (DPM) by Felzenszwalb et al. (2008) attempts to break down the complex appearance of objects into easier parts for training SVMs with latent structure variables which represent the model configuration and need to be inferred at training time. They use a coarse global template covering the entire object and higher resolution part templates to model the appearance of each part as illustrated in Figure 6. All templates are represented using HOG features. In addition, they generalize SVMs to handle latent variables such as the part position location.

An alternative to this representation is the Implicit Shape Model proposed by Leibe et al. (2008a) which learns a highly flexible representation of object shape. They extract local features around interest points and perform clustering to build up a codebook of local appearances that are characteristic for the particular object class under consideration. Based on this codebook, they learn where on the object the codebook entries may occur.

While the part-based models presented so far have been very successful, they can not represent contextual information which is necessary for occlusion reasoning. Usually, a separate context model is learned to handle occlusions, see Hoiem et al. (2008); Tu & Bai (2010); Desai et al. (2011); Yang et al. (2012). And-Or models embed a grammar to represent large structural and appearance variations in a reconfigurable hierarchy. Wu et al. (2016a) propose to learn an And-Or model which takes into account structural and appearance variations at multi-car, single-car and part-levels jointly to represent both context and occlusions.

5.1. 2D Object Detection

KITTI Geiger et al. (2012b) is among the most popular benchmarks for object detection systems in the autonomous car context. A similar popularity for the pedestrian detection task has the Caltech-USA dataset (Dollár et al. (2012)). In this work we would like to focus our attention on the KITTI benchmark since it allows us to compare object and pedestrian detection systems on the same data. We refer the interested reader to the survey papers (Benenson et al. (2014); Zhang et al. (2016b)) for an in-depth comparison of pedestrian detection systems on Caltech-USA. In Table 1 we show the state-of-the-art on the KITTI benchmark for object, pedestrian and cyclist detection from images. Note that for all result tables in this paper, we list only public methods which have a paper associated with them as the details for the anonymous entries cannot be discussed yet. The performance is assessed for three level of difficulties using PASCAL VOC intersection-over-union (IOU) (Everingham et al. (2010)). Easy examples have a minimum bounding box height of 40 px and are fully visible, whereas moderate examples have a minimum height of 25 px including partial occlusion and hard examples have the same minimum height but includes the maximum occlusion level. In Table 2 the estimation of the object's orientation is evaluated using the average orientation similarity (AOS) proposed in Geiger et al. (2012b).

Method	Moderate	Easy	Hard	Runtime
SubCNN – Xiang et al. (2016)	89.04 %	90.81 %	79.27 %	2 s / GPU
MS-CNN – Cai et al. (2016)	89.02 %	90.03 %	76.11 %	0.4 s / GPU
SDP+RPN – Yang et al. (2016)	88.85 %	90.14 %	78.38 %	0.4 s / GPU
Mono3D – Chen et al. (2016a)	88.66 %	92.33 %	78.96 %	4.2 s / GPU
3DOP – Chen et al. (2015c)	88.64 %	93.04 %	79.10 %	3s / GPU
MV3D (LIDAR + MONO) – Chen et al. (2016c)	87.67 %	89.11 %	79.54 %	0.45 s / GPU
SDP+CRC (ft) – Yang et al. (2016)	83.53 %	90.33 %	71.13 %	0.6 s / GPU
Faster R-CNN – Ren et al. (2015)	81.84 %	86.71 %	71.12 %	2 s / GPU
AOG – Wu et al. (2016a)	75.94 %	84.80 %	60.70 %	3 s / 4 cores
3DVP – Xiang et al. (2015b)	75.77 %	87.46 %	65.38 %	40 s / 8 cores
LSVM-MDPM-sv – Felzenszwalb et al. (2010)	56.48 %	68.02 %	44.18 %	10 s / 4 cores
ACF – Dollár et al. (2014)	54.74 %	55.89 %	42.98 %	0.2 s / 1 core

(a) KITTI Car Detection Leaderboard

Method	Moderate	Easy	Hard	Runtime
MS-CNN – Cai et al. (2016)	73.70 %	83.92 %	68.31 %	0.4 s / GPU
SubCNN – Xiang et al. (2016)	71.33 %	83.28 %	66.36 %	2 s / GPU
IVA – Zhu et al. (2016)	70.70 %	83.63 %	64.67 %	0.4 s / GPU
SDP+RPN – Yang et al. (2016)	70.16 %	80.09 %	64.82 %	0.4 s / GPU
3DOP – Chen et al. (2015c)	67.47 %	81.78 %	64.70 %	3s / GPU
Mono3D – Chen et al. (2016a)	66.68 %	80.35 %	63.44 %	4.2 s / GPU
Faster R-CNN – Ren et al. (2015)	65.90 %	78.86 %	61.18 %	2 s / GPU
SDP+CRC (ft) – Yang et al. (2016)	64.19 %	77.74 %	59.27 %	0.6 s / GPU
RPN+BF – Zhang et al. (2016a)	61.29 %	75.45 %	56.08 %	0.6 s / GPU
DPM-VOC+VP – Pepik et al. (2015)	44.86 %	59.48 %	40.37 %	8 s / 1 core
SubCat – Ohn-Bar & Trivedi (2015)	42.34 %	54.67 %	37.95 %	1.2 s / 6 cores
ACF – Dollár et al. (2014)	39.81 %	44.49 %	37.21 %	0.2 s / 1 core
LSVM-MDPM-sv – Felzenszwalb et al. (2010)	39.36 %	47.74 %	35.95 %	10 s / 4 cores

(b) KITTI Pedestrian Detection Leaderboard

Method	Moderate	Easy	Hard	Runtime
MS-CNN – Cai et al. (2016)	75.46 %	84.06 %	66.07 %	0.4 s / GPU
SDP+RPN – Yang et al. (2016)	73.74 %	81.37 %	65.31 %	0.4 s / GPU
SubCNN – Xiang et al. (2016)	71.06 %	79.48 %	62.68 %	2 s / GPU
3DOP – Chen et al. (2015c)	68.94 %	78.39 %	61.37 %	3s / GPU
IVA – Zhu et al. (2016)	67.47 %	80.17 %	59.66 %	0.4 s / GPU
Mono3D – Chen et al. (2016a)	66.36 %	76.04 %	58.87 %	4.2 s / GPU
Faster R-CNN – Ren et al. (2015)	63.35 %	72.26 %	55.90 %	2 s / GPU
SDP+CRC (ft) – Yang et al. (2016)	61.31 %	74.08 %	53.97 %	0.6 s / GPU
Regionlets – Wang et al. (2015)	58.72 %	70.41 %	51.83 %	1 s / > 8 cores
DPM-VOC+VP – Pepik et al. (2015)	31.08 %	42.43 %	28.23 %	8 s / 1 core
LSVM-MDPM-us – Felzenszwalb et al. (2010)	29.88 %	38.84 %	27.31 %	10 s / 4 cores

(c) KITTI Cyclist Detection Leaderboard

Table 1: **KITTI Object Detection Leaderboard.** Only image-based methods are shown in these tables, i.e., no laser scan data is used. The numbers represent average precision at different levels of difficulty based on the object size and the level of occlusion/truncation. Higher numbers indicate better performance.

Convolutional Neural Networks allowed a significant improvement in the performance of object detectors. In the beginning, CNNs were integrated in sliding-window approaches (Sermanet et al. (2013)). However, the precise localization of objects is challenging because of the large receptive fields and strides. Girshick et al. (2014), on the other hand, propose R-CNNs to solve the CNN localization problem with a "recognition using regions" paradigm. They generate many region proposals using selective search (Uijlings et al. (2013)), extract a fixed-length feature vector for each proposal using a CNN and classify each region with a linear SVM. Region-based CNNs are computationally expensive but several improvements have been proposed to reduce the computational burden (He et al. (2014); Girshick (2015)). He et al. (2014) use spatial pyramid pooling which allows to compute a convolutional feature map for the entire image with only one run of the CNN in contrast to R-CNN that needs to be applied on many image regions. Girshick (2015) further improve with a single-stage training algorithm that jointly learns to classify object proposals and refine their spatial locations. Even though these region-based networks have proven to be very successful on the PASCAL VOC benchmark, they could not achieve similar perfor-

Method	Moderate	Easy	Hard	Runtime
SubCNN – Xiang et al. (2016)	88.62 %	90.67 %	78.68 %	2 s / GPU
Mono3D – Chen et al. (2016a)	86.62 %	91.01 %	76.84 %	4.2 s / GPU
3DOP – Chen et al. (2015c)	86.10 %	91.44 %	76.52 %	3s / GPU
3DVP – Xiang et al. (2015b)	74.59 %	86.92 %	64.11 %	40 s / 8 cores
SubCat – Ohn-Bar & Trivedi (2015)	74.42 %	83.41 %	58.83 %	0.7 s / 6 cores
OC-DPM – Pepik et al. (2013)	64.42 %	73.50 %	52.40 %	10 s / 8 cores
AOG-View – Li et al. (2014)	63.31 %	76.70 %	50.34 %	3 s / 1 core
DPM-VOC+VP – Pepik et al. (2015)	61.84 %	72.28 %	46.54 %	8 s / 1 core
LSVM-MDPM-sv – Felzenszwalb et al. (2010)	55.77 %	67.27 %	43.59 %	10 s / 4 cores

(a) KITTI Car Detection and Orientation Estimation Leaderboard

Method	Moderate	Easy	Hard	Runtime
SubCNN – Xiang et al. (2016)	66.28 %	78.45 %	61.36 %	2 s / GPU
3DOP – Chen et al. (2015c)	59.80 %	72.94 %	57.03 %	3s / GPU
Mono3D – Chen et al. (2016a)	58.15 %	71.15 %	54.94 %	4.2 s / GPU
DPM-VOC+VP – Pepik et al. (2015)	39.83 %	53.55 %	35.73 %	8 s / 1 core
LSVM-MDPM-sv – Felzenszwalb et al. (2010)	35.49 %	43.58 %	32.42 %	10 s / 4 cores
SubCat – Ohn-Bar & Trivedi (2015)	34.18 %	44.32 %	30.76 %	1.2 s / 6 cores
RPN+BF – Zhang et al. (2016a)	32.55 %	40.91 %	29.52 %	0.6 s / GPU
ACF – Dollár et al. (2014)	28.46 %	35.69 %	26.18 %	1 s / 1 core

(b) KITTI Pedestrian Detection and Orientation Estimation Leaderboard

Method	Moderate	Easy	Hard	Runtime
SubCNN – Xiang et al. (2016)	63.65 %	72.00 %	56.32 %	2 s / GPU
3DOP – Chen et al. (2015c)	58.68 %	70.13 %	52.35 %	3s / GPU
Mono3D – Chen et al. (2016a)	54.97 %	65.56 %	48.77 %	4.2 s / GPU
DPM-VOC+VP – Pepik et al. (2015)	23.17 %	30.52 %	21.58 %	8 s / 1 core
LSVM-MDPM-sv – Felzenszwalb et al. (2010)	22.07 %	27.54 %	21.45 %	10 s / 4 cores

(c) KITTI Cyclist Detection and Orientation Estimation Leaderboard

Table 2: **KITTI Detection and Orientation Estimation Leaderboard.** Only image-based methods are shown in these tables, i.e., no laser scan data is used. The numbers represent average orientation similarity as described in Geiger et al. (2012b). Higher numbers indicate better detection and orientation estimation.

mance on KITTI. The main reason for this is that the KITTI dataset contains objects at many different scales and small objects which are often heavily occluded or truncated. These objects are hard to detect using the region-based networks. Therefore, several methods for obtaining better object proposals have been proposed (Ren et al. (2015); Chen et al. (2016b,a); Yang et al. (2016); Cai et al. (2016)).

Ren et al. (2015) have introduced Region Proposal Networks (RPN) in which the region proposal network shares full-image convolutional features with the detection network and thus doesn't increase computational costs. RPNs are trained end-to-end to generate high quality region proposals which are classified using the Fast R-CNN detector (Girshick (2015)). Chen et al. (2015c) use 3D information estimated from a stereo camera pair to extract better bounding box proposals. They place 3D candidate boxes on the ground plane and score them using 3D point cloud features. Finally, a CNN exploiting contextual information and using a multi-task loss jointly regresses the object's coordinates and orientation. Inspired by this approach, Chen et al. (2016a) learn to generate class-specific 3D object proposals for monocular images, exploiting contextual models as well as semantics. They generate proposals by exhaustively placing 3D bounding boxes on the ground plane and scoring

them with a standard CNN pipeline (Chen et al. (2015c)). Both methods Chen et al. (2015c) and Chen et al. (2016a) achieve comparable results to the best performing method in all detection task while outperforming all other methods on easy examples of KITTI car (Table 1a). In addition, they are among the best performing methods for the orientation estimation (Table 2).

An alternative approach is presented by Yang et al. (2016). In case of small objects a strong activation of convolutional neurons is more likely to occur in earlier layers. Therefore, Yang et al. (2016) use scale-dependent pooling which allows to represent a candidate bounding box using the convolutional features from the corresponding scale. In addition, they propose layer-wise cascaded rejection classifiers, treating convolutional features in early layers as weak classifiers, to efficiently eliminate negative object proposals. The proposed scale-dependent pooling approach is one of the best performing methods in all tasks (Table 1).

Cai et al. (2016) propose a multi-scale CNN consisting of a proposal sub-network and a detection sub-network. The proposal network, illustrated in Figure 7, performs detection at multiple output layers and these complementary scale-specific detectors are combined to produce a strong multi-scale object
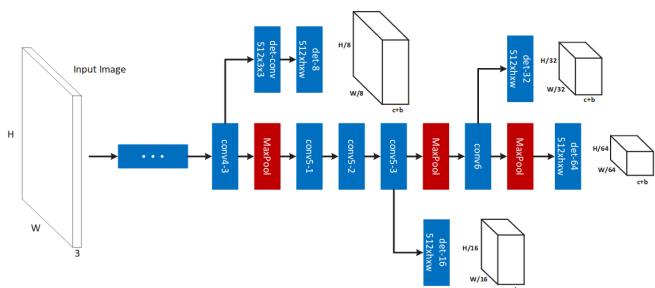
Figure 7: The proposal sub-network presented by Cai et al. (2016) performs detection at multiple output layers to match objects at different scales. Scale-specific detectors are combined to produce a strong multi-scale object detector. Adapted from Cai et al. (2016).

Figure 8: The network proposed by Chen et al. (2016b) combines region-wise features from the bird's eye view, the front view of the LiDAR point cloud as well as the RGB image as input for a deep fusion network. Adapted from Chen et al. (2016b).

detector. Their multi-scale CNN outperforms all other methods on KITTI pedestrian and cyclist (Tables 1b,1c) while ranking second on KITTI car (Table 1a). Xiang et al. (2016) propose a region proposal network that uses subcategory information obtained from 3DVP (Xiang et al. (2015b)), to guide the proposal generating process, and a detection network for joint detection and subcategory classification. Object subcategories are defined for objects with similar properties or attributes such as appearance, pose or shape. The subcategory information allows them to outperform all other methods for the detection task on KITTI cars (Table 1a) and to achieve the best performance in the orientation estimation (Table 2).

5.2. 3D Object Detection from 2D Images

Geometric 3D representations of object classes can recover far more details than just 2D or 3D bounding boxes, however most of today's object detectors are focused on robust 2D matching. Zia et al. (2013) exploit the fact that high-quality 3D CAD models are available for many important classes. From these models, they obtain coarse 3D wireframe models using principal components analysis and train detectors for the vertices of the wireframe. At test time, they generate evidence for vertices by densely applying the detectors. Zia et al. (2015) extend this work by directly using detailed 3D CAD models in their formulation, combining them with explicit representations of likely occlusion patterns. Further, a ground plane is jointly estimated to stabilize the pose estimation process. This extension outperforms the pseudo-3D model of Zia et al. (2013) and shows the benefits of reasoning in true metric 3D space.

While these 3D representations provide more faithful descriptions of objects they can not yet compete with state-of-the-art detectors using 2D bounding boxes. To overcome this problem, Pepik et al. (2015) propose a 3D extension of the powerful deformable part model (Felzenszwalb et al. (2008)), which combines the 3D geometric representation with robust matching to real-world images. They further add 3D CAD information of the object class of interest as geometry cue to enrich the appearance model.

5.3. 3D Object Detection from 3D Point Clouds

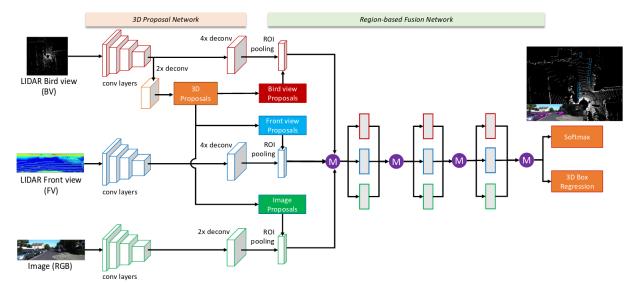The KITTI dataset Geiger et al. (2012b) provides synchronized camera and LiDAR frames and allows the comparison of

image-based and LiDAR-based approaches on the same data. In contrast to cameras, LiDAR laser range sensors directly provide accurate 3D information which simplifies the extraction of object candidates and can be helpful for the classification task as it provides 3D shape information. However, 3D data from laser scanners is typically sparse and its spatial resolution is limited. Therefore, the state-of-the-art relying only on laser range data can not reach the performance of camera-based detection systems, yet. In Table 3 we show the LiDAR-based state-of-the-art on the KITTI benchmark for object, pedestrian and cyclist detection. The performance is assessed similar to the image-based approaches using the PASCAL intersection-over-union by projecting the 3D bounding boxes into the image plane.

Wang & Posner (2015) propose an efficient scheme to apply the common 2D sliding window detection approach to 3D data. More specifically, they exploit the sparse nature of the problem using a voting scheme to search all possible object locations and orientations. Li et al. (2016b) improve upon these results by exploiting a fully convolutional neural network for detecting vehicles from range data. They represent the data in a 2D point map, and predict an objectness confidence and a bounding box simultaneously using a single 2D CNN. The encoding used to represent the data allows them to predict the full 3D bounding box of the vehicles. Engelcke et al. (2016) leverage a feature-centric voting scheme to implement a novel convolutional layer which exploits the sparsity of the point cloud. Additionally, they propose to use the L_1 penalty for regularization.

Relying on laser range data alone makes the detection task challenging due to the limited density of the laser scans. Thus, existing LiDAR-based approaches perform weaker compared to their image-based counterparts on the KITTI datasets. Chen et al. (2016c) combine LiDAR laser range data with RGB images for object detection. In their approach, the sparse point cloud is encoded using a compact multi-view representation and a proposal generation network utilizes the bird's eye view representation of the point cloud to generate 3D candidates. Finally, they combine region-wise features from multiple views with a deep fusion scheme as illustration in Figure 8. This approach outperforms the other LiDAR-based approaches by a significant margin and achieves state-of-the art performance in the KITTI car benchmarks (Tables 1a,3a).

14

Method	Moderate	Easy	Hard	Runtime
MV3D (LIDAR + MONO) – Chen et al. (2016c)	87.67 %	89.11 %	79.54 %	0.45 s / GPU
MV3D (LIDAR) – Chen et al. (2016c)	79.24 %	87.00 %	78.16 %	0.3 s / GPU
MV-RGBD-RF – González et al. (2015)	69.92 %	76.40 %	57.47 %	4 s / 4 cores
Vote3Deep – Engelcke et al. (2016)	68.24 %	76.79 %	63.23 %	1.5 s / 4 cores
VeloFCN – Li et al. (2016b)	53.59 %	71.06 %	46.92 %	1 s / GPU
Vote3D – Wang & Posner (2015)	47.99 %	56.80 %	42.57 %	0.5 s / 4 cores
CSoR – Plotkin (2015)	26.13 %	34.79 %	22.69 %	3.5 s / 4 cores
mBoW – Behley et al. (2013)	23.76 %	36.02 %	18.44 %	10 s / 1 core

(a) KITTI Car Detection Leaderboard

Method	Moderate	Easy	Hard	Runtime
MV-RGBD-RF – González et al. (2015)	56.59 %	73.30 %	49.63 %	4 s / 4 cores
Vote3Deep – Engelcke et al. (2016)	55.37 %	68.39 %	52.59 %	1.5 s / 4 cores
Fusion-DPM – Premebida et al. (2014)	46.67 %	59.51 %	42.05 %	30 s / 1 core
Vote3D – Wang & Posner (2015)	35.74 %	44.48 %	33.72 %	0.5 s / 4 cores
mBoW – Behley et al. (2013)	31.37 %	44.28 %	30.62 %	10 s / 1 core

(b) KITTI Pedestrian Detection Leaderboard

Method	Moderate	Easy	Hard	Runtime
Vote3Deep – Engelcke et al. (2016)	67.88 %	79.92 %	62.98 %	1.5 s / 4 cores
MV-RGBD-RF – González et al. (2015)	42.61 %	52.97 %	37.42 %	4 s / 4 cores
Vote3D – Wang & Posner (2015)	31.24 %	41.43 %	28.60 %	0.5 s / 4 cores
mBoW – Behley et al. (2013)	21.62 %	28.00 %	20.93 %	10 s / 1 core

(c) KITTI Cyclist Detection Leaderboard

Table 3: **KITTI LiDAR Detection Leaderboard.** Methods that focus on LiDAR scans and methods combining LiDAR with RGB images are presented. The numbers represent average precision at different levels of difficulty. Higher numbers indicate better performance.

5.4. *Person Detection*

While so far we have discussed general object detection algorithms, we now focus on specific approaches to person or pedestrian detection which are of high relevance to any autonomous system interacting with a real environment. As human behavior is less predictable than the behavior of a car, reliable person detection is necessary to drive safely in the proximity of pedestrians. The detection of people is particularly difficult because of the large variety of appearances due to different clothing and articulated poses. Furthermore, the articulation and interaction of pedestrians can strongly affect the appearance of pedestrians in case of partial occlusion.

Pedestrian Protection Systems: This problem has been deeply investigated for advanced driver assistance systems to increase road safety. Pedestrian protection systems (PPS) detect the presence of stationary and moving people around a moving vehicle in order to warn the driver against dangerous situations. Even though missed detections of a PPS can still be handled by the driver, the pedestrian detection of an autonomous car needs to be flawless. The pedestrian detection system needs to be robust against all weather conditions and efficient for real-time detection. Geronimo et al. (2010) survey pedestrian detection for Advanced Driver Assistance Systems.

Surveys: Enzweiler & Gavrila (2009) give a very broad overview of different architectures for monocular pedestrian detection. They make the observation that the HOG/SVM combination as proposed by Dalal & Triggs (2005) works well at higher res-

olutions with higher processing time whereas AdaBoost cascade approaches are superior at lower resolutions, achieving near real-time performance. In their survey, Benenson et al. (2014) found no clear evidence that a certain type of classifier (e.g., SVM or decision forests) is better suited than others. In particular, Wojek & Schiele (2008b) show that AdaBoost and linear SVM perform roughly the same if enough features are given. Moreover, Benenson et al. (2014) observe that part based models like (Felzenszwalb et al. (2008)) improve results only slightly compared to the much simpler approach of Dalal & Triggs (2005). They conclude that the number and diversity of features is clearly an important factor for the performance of classifiers since the classification problem becomes easier with higher dimensional representations. Consequently, today all state-of-the-art pedestrian detection systems use convolutional neural networks and learn feature representations in an end-to-end fashion (Cai et al. (2016); Xiang et al. (2016); Zhu et al. (2016); Yang et al. (2016); Chen et al. (2015c); Ren et al. (2015)).

Temporal Cues: Similarly, Shashua et al. (2004) point out the importance of good features for the person detection task. They noted that the integration of additional cues measured over time (dynamic gait, motion parallax) and situation specific features (such as leg positions at certain poses) are key for reliable detection. Wojek et al. (2009) notice that most pedestrian detection systems rely only on a single image as input and do not exploit the available temporal information of objects in video sequences. They show significant improvement in detection

15

performance by incorporating motion cues and combining different complementary feature types.

Scarcity of Target Class: The enlargement of training data allows to train sophisticated models for the detection problem. However, the generation of examples belonging to the target class is usually time consuming because of manual labeling while many negative examples can be easily obtained. Enzweiler & Gavrila (2008) address the bottleneck caused by the scarcity of samples of the target class. They create synthesized virtual samples with a learned generative model to enhance a discriminative model. The generative model captures prior knowledge about the pedestrian class and allows significant improvement in the classification performance.

Real-time Pedestrian Detection: In case of a potential collision with pedestrians a fast detection allows early intervention of the autonomous system. Benenson et al. (2012) provide fast and high quality pedestrian detections based on better handling of scales and exploiting depth extracted from stereo. Instead of resizing the images, they scale HOG features similar to Viola & Jones (2004). The Stixel World representation (Badino et al. (2009)) provides depth information which allows to significantly reduce the search space and detect pedestrians at 80 Hz in a parallel framework.

5.5. Human Pose Estimation

The pose and gaze of a person provides important information to the autonomous vehicle about the behavior and intention of the person. However, the pose estimation problem is challenging since the pose space is very large and typically people can only be observed on low resolutions, because of their size and distance to the vehicle. Several approaches have been proposed to jointly estimate the pose and body parts of a person. Traditionally, a two-staged approach was used by first detecting body parts and then estimating the pose as in (Pishchulin et al. (2012); Gkioxari et al. (2014); Sun & Savarese (2011)). This is problematic in cases when people are in proximity of each other because body-parts can be wrongly assigned to different instances.

Pishchulin et al. (2016) present DeepCut, a model which jointly estimates the poses of all people in an image. The formulation is based on partitioning and labeling a set of body-part hypotheses obtained from a CNN-based part detector. The model jointly infers the number of people, their poses, spatial proximity and part level occlusions. Bogo et al. (2016) use DeepCut to estimate the 3D pose and 3D shape of a human body from a single unconstrained image. SMPL, a 3D body shape model proposed by Loper et al. (2015), is fit to predictions of the 2D body joint locations from DeepCut. SMPL captures correlations in human shape across the population which allows to robustly fit human poses even in the presence of weak observations.

5.6. Discussion

Object detection works already quite well in case of high resolution with little occlusions. For the easy and moderate cases of the car detection task (Table 1a) many methods show impressive performance. The pedestrian and cyclist detection task (Tables 1b,1c) is more challenging and thus weaker overall performance can be observed. One reason for this is the limited number of training examples and the possibility of confusing cyclists and pedestrians which differ only via their context and semantics. Remaining major problems across tasks are detection of small objects and highly occluded objects. In the leaderboards this manifests in a significant drop in performance when comparing easy, moderate and hard examples. Qualitatively, this can be observed in Figures 9, 10,11 where we show typical estimation errors of the best performing methods on the KITTI dataset. A major source of errors are crowds of pedestrians, groups of cyclists and lines of cars that cause many occlusions and lead to missing detections for all methods. Furthermore, a large amount of distant objects needs to be detected in some cases which is still a challenging task for modern methods since the amount of information provided by these objects is very low.

(a) Images with Largest Number of True Positive Detections

(b) Images with Largest Number of False Positive Detections

(c) Images with Largest Number of False Negative Detections

Figure 9: **KITTI Vehicle Detection Analysis.** Each figure shows images with a large number of true positive (TP) detections, false positive (FP) detections and false negative (FN) detections, respectively. If all detectors agree on TP, FP or FN, the object is marked in red. If only some of the detectors agree, the object is marked in yellow. The ranking has been established by considering the 15 leading methods published on the KITTI evaluation server at time of submission.

(a) Images with Largest Number of True Positive Detections

(b) Images with Largest Number of False Positive Detections

(c) Images with Largest Number of False Negative Detections

Figure 10: **KITTI Pedestrian Detection Analysis.** Each figure shows images with a large number of true positive (TP) detections, false positive (FP) detections and false negative (FN) detections, respectively. If all detectors agree on TP, FP or FN, the object is marked in red. If only some of the detectors agree, the object is marked in yellow. The ranking has been established by considering the 15 leading methods published on the KITTI evaluation server at time of submission.

(a) Images with Largest Number of True Positive Detections

(b) Images with Largest Number of False Positive Detections

(c) Images with Largest Number of False Negative Detections

Figure 11: **KITTI Cyclist Detection Analysis.** Each figure shows images with a large number of true positive (TP) detections, false positive (FP) detections and false negative (FN) detections, respectively. If all detectors agree on TP, FP or FN, the object is marked in red. If only some of the detectors agree, the object is marked in yellow. The ranking has been established by considering the 15 leading methods published on the KITTI evaluation server at time of submission.
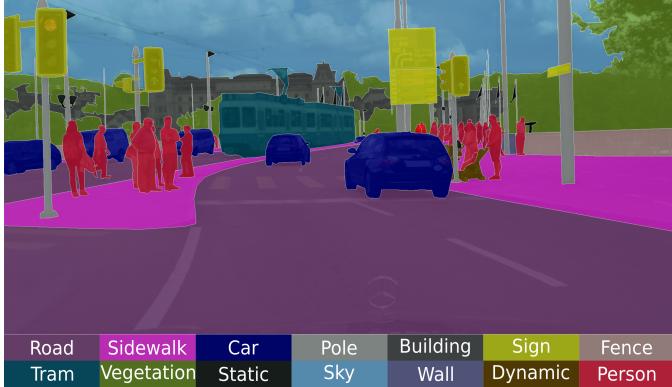
19

| Road | Sidewalk | Car | Pole | Building | Sign | Fence |
| Tram | Vegetation | Static | Sky | Wall | Dynamic | Person |

Figure 12: Semantic segmentation of a scene from the Cityscapes dataset by Cordts et al. (2016) recorded in Zürich.

6. Semantic Segmentation

Semantic segmentation, is a fundamental topic in computer vision. The goal of semantic segmentation is to assign each pixel in the image a label from a predefined set of categories. The task is illustrated in Figure 12 with all pixel of a certain category colorized in as specific color in a scene of the Cityscapes dataset[25] by Cordts et al. (2016) recorded in Zürich. Segmentation of images into semantic regions usually found in street scenes, such as cars, pedestrians, or road affords a comprehensive understanding of the surrounding which is essential to autonomous navigation. Challenges of semantic segmentation arise from the complexity of the scene and the size of the label space.

Formulation: Traditionally, the semantic segmentation problem was posed as maximum a posteriori (MAP) inference in a conditional random field (CRF), defined over pixels or super-pixels (He et al. (2004, 2006)). However, these early formulations were not efficient and could only handle only datasets of limited size and a small number of classes. Furthermore, only very simple features such as color, edge and texture information have been exploited. Shotton et al. (2009) observed that more powerful features can significantly boost performance and proposed an approach based on a novel type of features called texture-layout filter that exploits the textural appearance of objects, its layout and textural context. They combine texture-layout filters with lower-level image features in a CRF to obtain pixel-level segmentations. Randomized boosting and piecewise training techniques are exploited to efficiently train the model.

Hierarchical and long-range connectivity as well as higher-order potentials defined on image regions were considered to tackle the limited ability of CRFs to model long-range interactions within the image. However, methods based on image regions (He et al. (2004); Kumar & Hebert (2005); He et al. (2006); Kohli et al. (2009); Ladicky et al. (2009, 2014)) are restricted by the accuracy of the image segmentations used as input. In contrast, Krähenbühl & Koltun (2011) propose a highly efficient inference algorithm for fully connected CRF models which models pairwise potentials between all pairs of pixels in the image.

The methods so far consider each object class independently while the co-occurrence of object classes can be an important clue for semantic segmentation, as for example cars are more likely to occur in a street scene than in an office. Consequently, Ladicky et al. (2010) propose to incorporate object class co-occurrence as global potentials in a CRF. They show how these potentials can be efficiently optimized using a graph cut algorithm and demonstrate improvements over simpler pairwise models.

The success of deep convolutional neural networks for image classification and object detection has sparked interest in leveraging their power for solving the pixel-wise semantic segmentation task. The fully convolutional neural network (Long et al. (2015)) is one of the earliest works which applies CNNs to the image segmentation problem. However, while modern convolutional neural networks for image classification combine multi-scale contextual information by consecutive pooling and subsampling layers that lower the resolution, semantic segmentation requires multi-scale contextual reasoning together with full-resolution dense prediction. In the following we will review recent approaches which address this problem.

We focus the comparison of different semantic segmentation approaches on the Cityscapes dataset[26] by Cordts et al. (2016) described in Section 2 because of the autonomous driving context. Table 4a shows the leaderboard of Cityscapes for the pixel-level semantic labeling task. The intersection-over-union metric is provided for two semantic granularities, i.e., classes and categories, and additionally the instance-weighted IoU is reported for both granularities to penalize methods ignoring small instances.

Structured CNNs: Recently, several methods have been proposed to tackle the opposing needs of multi-scale inference and full-resolution prediction output. Dilated convolutions have been proposed (Chen et al. (2015b); Yu & Koltun (2016)) to enlarge the receptive field of neural networks without loss of resolution. Their operation corresponds to regular convolution with dilated filters which allows for efficient multi-scale reasoning while limiting the increase in the number of model parameters.

In the SegNet model, Badrinarayanan et al. (2015) have replaced the traditional decoder in a deep architecture with a network which consists of a hierarchy of decoders one corresponding to each encoder. Each decoder maps a low resolution feature map of an encoder (max-pooling layer) to a higher resolution feature map. In particular, the decoder in their model takes advantage of the pooling indices computed in the max-pooling step of the corresponding encoder to implement the upsampling process. This eliminates the need to learn the upsampling and thus results in a smaller number of parameters. Furthermore, sharper segmentation boundaries have been demonstrated using this approach.

While activation maps at lower-levels of the CNN hierarchy lack object category specificity, they do contain higher spa-

---

[25]https://www.cityscapes-dataset.com/

[26]https://www.cityscapes-dataset.com/

Method	IoU class	iIoU class	IoU category	iIoU category
ResNet-38 – Wu et al. (2016b)	80.6	57.8	91	79.1
PSPNet – Zhao et al. (2016)	80.2	58.1	90.6	78.2
RefineNet – Lin et al. (2016a)	73.6	47.2	87.9	70.6
LRR-4x – Ghiasi & Fowlkes (2016)	71.8	47.9	88.4	73.9
FRRN – Pohlen et al. (2016)	71.8	45.5	88.9	75.1
Adelaide_context – Lin et al. (2016b)	71.6	51.7	87.3	74.1
DeepLabv2-CRF – Chen et al. (2016b)	70.4	42.6	86.4	67.7
Dilation10 – Yu & Koltun (2016)	67.1	42	86.5	71.1
DPN – Liu et al. (2015)	66.8	39.1	86	69.1
Scale invariant CNN + CRF – Krešo et al. (2016)	66.3	44.9	85	71.2
FCN 8s – Long et al. (2015)	65.3	41.7	85.7	70.1
DeepLab LargeFOV StrongWeak – Papandreou et al. (2015)	64.8	34.9	81.3	58.7
Pixel-level Encoding for Instance Segmentation – Uhrig et al. (2016)	64.3	41.6	85.9	73.9
DeepLab LargeFOV Strong – Chen et al. (2015b)	63.1	34.5	81.2	58.7
Segnet basic – Badrinarayanan et al. (2015)	57	32	79.1	61.9

(a) CITYSCAPES Semantic Segmentation Leaderboard

Method	AP	AP 50%	AP 100m	AP 50m
DIN – Arnab & Torr (2017)	20	38.8	32.6	37.6
Shape-Aware Instance Segmentation – Hayder et al. (2016)	17.4	36.7	29.3	34
DWT – Bai & Urtasun (2016)	15.6	30	26.2	31.8
InstanceCut – Kirillov et al. (2016)	13	27.9	22.1	26.1
Joint Graph Decomposition and Node Labeling – Levinkov et al. (2016)	9.8	23.2	16.8	20.3
Pixel-level Encoding for Instance Segmentation – Uhrig et al. (2016)	8.9	21.1	15.3	16.7
R-CNN + MCG convex hull – Cordts et al. (2016)	4.6	12.9	7.7	10.3

(b) CITYSCAPES Instance Segmentation Leaderboard

Table 4: **CITYSCAPES Semantic and Instance Segmentation Leaderboards.** Segmentation performance is measured by class intersection-over-union and instance-level intersection-over-union. Instance detection performance is measured in terms of several average precision variants. See also Cordts et al. (2016).
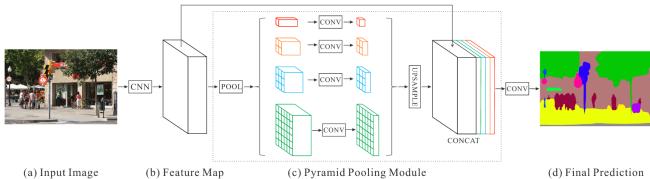

Figure 13: Overview of the method proposed by Zhao et al. (2016). The pyramid parsing module (c) is applied on a CNN feature map (b) and fed into a convolutional layer for pixel-level estimation (d). Adapted from Zhao et al. (2016).

tial resolution information. Ghiasi & Fowlkes (2016) leverage this assumption and propose to construct a Laplacian pyramid based on a fully convolutional network. Aggregating information at multiple scales allows them to successively refine the boundary reconstructed from lower-resolution layers. They achieve this by using skip connections from higher resolution feature maps and multiplicative confidence gating, penalizing noisy high-resolution outputs in regions where the low-resolution predictions have high confidence. With this approach Ghiasi & Fowlkes (2016) achieve competitive results on Cityscapes Table 4a.

One of the best performing methods on Cityscapes was proposed by Zhao et al. (2016) using a pyramid scene parsing network, illustrated in Figure 13, to incorporate global context information into the pixel-level prediction task. Specifically, they apply a pyramid parsing module to the last convolutional layer of a CNN which fuses features of several pyramid scales to combine local and global context information. The resulting representation is fed into a convolution layer to obtain final per-pixel predictions.

Simonyan & Zisserman (2015) and Szegedy et al. (2015) have shown that the depth of a CNN is crucial to represent rich features. However, increasing the depth of a network lead to the saturation and degradation of the accuracy. He et al. (2016) propose deep residual learning framework (ResNet) to address this problem. They let each stacked layer learn a residual mapping instead of the original, unreferenced mapping. This allows them to train deeper networks with improving accuracy while plain networks (simply stacked networks) exhibited higher training errors. Pohlen et al. (2016) present a ResNet-like architecture that provides strong recognition performance while preserving high-resolution information throughout the entire network by combining two different processing streams. One stream passes through a sequence of pooling layers, whereas the other stream processes feature maps at full image resolution. The two processing streams are combined at the full image resolution using residuals. Wu et al. (2016b) have proposed a more efficient ResNet architecture by analyzing the effective depths of

residual units. They point out that ResNets behave as linear ensembles of shallow networks. Based on this understanding they design a group of relatively shallow convolutional networks for the task of semantic image segmentation. While Pohlen et al. (2016) achieve competitive results on Cityscapes (Table 4a), Wu et al. (2016b) outperform all others in all measures besides the instance-weighted class-level IoU.

Conditional Random Fields: A different way to address the needs of multi-scale inference and full resolution prediction is the combination of CNNs with CRF models. Chen et al. (2015b) propose to refine the label map obtained using a convolutional neural network using a fully connected CRF model (Krähenbühl & Koltun (2011)). The CRF allows to capture fine details based on the raw RGB input which are missing in the CNN output due to the limited spatial accuracy of the CNN model. In similar spirit, Jampani et al. (2016) generalize bilateral filters and unroll the CRF program which allows for end-to-end training of the (generalized) filter parameters from data. This effectively allows for reasoning over larger spatial regions within one convolutional layer by leveraging input features as a guiding signal.

Inspired by higher order CRFs for semantic segmentation, Gadde et al. (2016a) propose a new Bilateral Inception module for CNN architectures as an alternative to structured CNNs and CRF techniques. They use the assumption that pixels which are spatially and photometrically similar are more likely to have the same label. This allows them to directly learn long-range interactions, thereby removing the need for post-processing using CRF models. Specifically, the proposed modules propagate edge-aware information between distant pixels based on their spatial and color similarity, incorporating the spatial layout of superpixels. Propagation of information is achieved by applying bilateral filters with Gaussian kernels at various scales.

Discussion: The focus on multi-scale inference of recent methods led to impressive results in pixel-level semantic segmentation on Cityscapes. Today, the top methods in Cityscapes Table 4b reach an impressive IoU of almost 81% over classes and 91% over categories. In contrast, the instance-weighted IoU is always below 58% over classes and 80% over categories. This indicates that semantic segmentation works well with instances covering large image areas but is still problematic with instances covering small regions. Similarly to the detection in low resolutions discussed in Section 5.6, small regions provide only little information to assign the correct label. Furthermore segmenting out small, and possibly occluded objects is a challenging task which might require novel approaches to jointly perform depth estimation and depth-adaptive recognition.

6.1. Semantic Instance Segmentation

The goal of semantic instance segmentation is simultaneous detection, segmentation and classification of every individual object in an image. Unlike semantic segmentation, it provides information about the position, semantics, shape and count of individual objects, and therefore has many applications in autonomous driving. For the task of semantic instance segmen-

tation, there exist two major lines of research: Proposal-based and proposal-free instance segmentation.

In Table 4b we show the leaderboard of semantic instance segmentation methods on the Cityscapes dataset. The performance is assessed with the average precision on the region level averaged across a range of overlap thresholds (AP), for an overlap value of 50 % (AP 50%) and for objects within 100 m and 50 m (AP 100m, AP 50m).

Proposal-based Instance Segmentation: Proposal-based instance segmentation methods extract class-agnostic proposals which are classified as an instance of a certain semantic class in order to obtain pixel-level instance masks. Region proposals like Multiscale Combinatorial Grouping (Arbeláez et al. (2014)) can be directly used as instance segments. Coarser representations such as bounding boxes need further refinement to obtain the instance mask. Unfortunately, proposal-based algorithms are slow at inference time due to the computationally expensive proposal generation step. To avoid this bottleneck, Dai et al. (2016) propose a fully convolutional network with three stages. They extract box proposals, use shared features to refine these to segments, and finally classify them into semantic categories. The causal relations between the outputs of the stages complicate training of the multi-task cascade. However, the authors show how these difficulties can be overcome using a differentiable layer which allows for training the whole model in an end-to-end fashion.

Proposal-based instance segmentation methods that use proposals in the form of bounding boxes to predict a binary segmentation mask are sensitive to errors in the proposal generation process including wrongly scaled or shifted bounding boxes. To tackle this problem, Hayder et al. (2016) present a new object representation. More specifically, they propose a shape aware object mask network that predicts a binary mask for each bounding box proposal, potentially extending beyond the box itself. They integrate the object mask network into the Multitask Network Cascade framework of Dai et al. (2016) by replacing the original mask prediction stage. The shape aware approach is the second best performing method on Cityscapes (Table 4b).

Proposal-free Instance Segmentation: Recently, a number of alternative methods to proposal-based instance segmentation have been proposed in the literature. These methods jointly infer the segmentation and the semantic category of individual instances by casting instance segmentation directly as a pixel labeling task.

Zhang et al. (2015, 2016c) train a fully convolutional neural networks (FCN) to directly predict pixel-level instance segmentation while the instance ID encodes a depth ordering. They improve the predictions and enforce consistency with a subsequent Markov Random Field. Uhrig et al. (2016) propose a method based on FCN to jointly predict semantic segmentation as well as depth and an instance-based direction relative to the centroid of each instance. The instance segmentation pipeline is illustrated in Figure 14. However, they require ground-truth depth data for training their model. Kirillov et al. (2016) present a proposal-free method which combines semantic segmentation
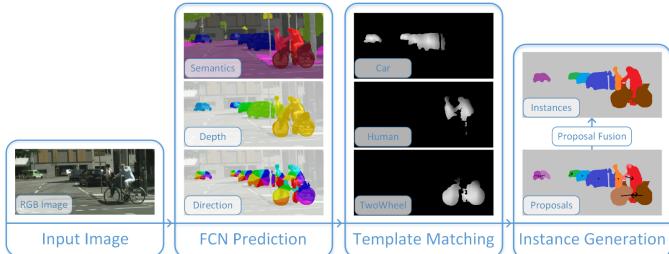
Figure 14: Uhrig et al. (2016) predict semantics, depth and instance center direction from the input image to compute template matching score maps for semantic categories. They fuse them after generating instance proposals to obtain an instance segmentation. Adapted from Uhrig et al. (2016).

and object boundary detection via global reasoning in a multi-cut formulation to infer semantic instance segmentation. Bai & Urtasun (2016) combine intuitions from classical watershed transform and deep learning to create an energy map where the basins corresponds to object instances. This allows them to cut at a single energy level to obtain an pixel-level instance segmentation. Kirillov et al. (2016) and Bai & Urtasun (2016) both achieve competitive results on Cityscapes (Table 4b). However, Arnab & Torr (2017) outperform all others by feeding an initial semantic segmentation into an instance subnetwork. Specifically, the initial category-level segmentation is used along cues from the output of an object detector within an end-to-end CRF to predict pixel-level instances.

Discussion: The instance segmentation task is much more difficult than the semantic segmentation task. Each instance need to be carefully annotated separately whereas in semantic segmentation groups of one semantic class can be annotated together when they occur next to each other. In addition, the number of instance varies greatly between different images. In the autonomous driving context often a wide view is present. Therefore, a large number of instances that appear are rather small in the image making them challenging to detect. In contrast to bounding boxes discussed in Section 5.6, the exact shape of each object instance needs to be inferred in this task. For these reasons, the state-of-the art is still struggling with the Cityscape dataset (Table 4b) reaching an average precision of 20% or less.

6.2. Label Propagation

Creating large scale image datasets with highly accurate pixel-level annotations is labor intensive, and thus very expensive to obtain the desired degree of quality. Semi-supervised methods for annotation of video sequences can help to reduce this cost. Compared to annotating individual images, video sequences offer the advantage of temporal consistency between consecutive frames. Label propagation techniques take advantage of this fact by propagating annotations from a small set of annotated keyframes to all unlabeled frames based on color information and motion estimates.

Towards this goal, Badrinarayanan et al. (2010) propose a coupled Bayesian network for joint modeling of the image sequence and pixel-wise labels. Specifically, they employ a prop-

agation scheme based on correspondences obtained from image patch based similarities and semantically consistent regions to transfer label information to unlabeled frames between annotated keyframes. Budvytis et al. (2010) extend this approach by proposing a hybrid model of the generative propagation introduced in Badrinarayanan et al. (2010) as well as a discriminative classification stage which tackles occlusions and disocclusions, and allows to propagate over larger time frames. To correct erroneous label propagation, Badrinarayanan et al. (2014) propose a superpixel based mixture-of-tree model for temporal correlation. Vijayanarasimhan & Grauman (2012) tackle the problem of selecting the most promising key frames for manual labeling such that the expected propagation error is minimized.

While the aforementioned methods transfer annotations in 2D, Chen et al. (2014); Xie et al. (2016) propose to annotate directly in 3D and then transfer these annotations into the image domain. Given a source of 3D information (e.g., stereo, laser), these approaches are able to produce improved semantic accuracy and time coherent labels while limiting annotation costs. Towards this goal, Chen et al. (2014) use annotations from KITTI (Geiger et al. (2013)) and leverage 3D car CAD models to infer separate figure-ground segmentations for all cars in the image. In contrast, Xie et al. (2016) reason jointly about all objects in the scene and also handle categories for which CAD models or 3D point measurements are unavailable. To this end, they propose a non-local CRF model which reasons jointly about semantic and instance labels of all 3D points and pixels in the image.

6.3. Semantic Segmentation with Multiple Frames

Semantic segmentation from movable platforms such as autonomous vehicles has become an active area of research due to the need of autonomous systems for recognizing their surrounding environment. As such systems are typically equipped with video cameras, temporal correlation between adjacent frames can be exploited to improve segmentation accuracy, efficiency and robustness.

Towards this goal, Floros & Leibe (2012) propose graphical models operating on video sequences in order to enforce temporal consistency between frames. Specifically, they have proposed a CRF where temporal consistency between consecutive video frames is ensured by linking corresponding image pixels to the inferred 3D scene points obtained by Structure from Motion (SfM). Compared to an image-only baseline they achieve an improved segmentation performance and observe a good generalization to varying image conditions.

3D reconstruction works relatively well for static scenes but is still an open problem in dynamic scenes. Feature-sensitive CRF models have been very successful in semantic image segmentation but the considered distance measure does not appropriately model spatio-temporal correspondences. The presence of both scene and camera motion makes temporal association in videos a challenging task. Because of the possibility of significant optical flow due to such motions, Euclidean distance in the space-time volume is not a good surrogate for correspondence. To tackle this problem, Kundu et al. (2016) propose a method
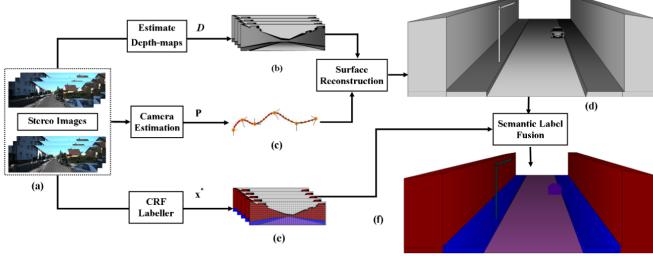
Figure 15: From a stereo image pair (a) Sengupta et al. (2013) compute the disparity map (b) and track the camera motion (c). They use both outputs to obtain a volumetric representation (d) and fuse the semantic segmentation of street images (e) into a 3D semantic model of the scene (f). Adapted from Sengupta et al. (2013).

for optimizing the feature space of a dense CRF for spatio-temporal regularization. Specifically, the feature space is optimized such that distances between features associated with corresponding points are minimized using correspondences from optical flow. The resulting mapping is exploited by the CRF to achieve long-range regularization over the entire video volume.

6.4. Semantic Segmentation of 3D Data

Autonomous systems need to recognize their surroundings to identify and interact with objects of interest. While the problem of semantic object labeling has been studied extensively, most of these algorithms work in the 2D image domain where each pixel in the image is labeled with a semantic category such as car, road or pavement. However, 2D images lack important information such as the 3D shape and scale of objects which are strong cues for object class segmentation and facilitate the detection and separation of individual object instances.

Sengupta et al. (2012) present an approach to generate a semantic overhead map of an urban scene from street level images. They formulate the problem using two CRFs. The first is used for semantic image segmentation of the street view images treating each image independently. Each street view image is then related by a geometrical function that back projects a region from the image into the overhead map. The outputs of this phase are then aggregated over many images to form the input for a second CRF producing a labeling of the ground plane. However, their method does not go beyond the flat world assumption to deliver dense semantic reconstruction using multiple street view images.

Towards this goal, Sengupta et al. (2013) propose an approach illustrated in Figure 15 where a dense semantic 3D reconstruction is generated using multiple street view images. They use visual odometry for ego-motion estimation according to which depth-maps generated from input stereo image pairs are fused. This allows them to generate a volumetric 3D representation of the scene. In parallel, input images are semantically classified using a CRF model. The results of segmentation are then aggregated across the sequence to generate the final 3D semantic model. However, the object labeling is performed in the image domain and then projected onto the model. As a result, these methods fail to fully exploit all structural constraints present in road scenes.

Valentin et al. (2013) tackle the problem of semantic scene reconstruction in 3D space by combining both structural and appearance cues. They use input depth estimates to generate a triangulated mesh representation of the scene and apply a cascaded classifier to learn geometric cues from the mesh and appearance cues from images. Subsequently, they solve for the labeling in 3D by defining a CRF over the scene mesh. However, they approach requires inference on the whole mesh an does not allow for incrementally adding information in an online setting as common in the autonomous driving context.

Hackel et al. (2016) propose a fast semantic segmentation approach for 3D point clouds with strongly varying densities. They construct approximate multi-scale neighborhoods by down-sampling the entire point cloud, to generate a multi-scale pyramid with decreasing density, and searching for the nearest neighbors per scale. This scheme allows to extract rich feature representation, that captures the geometry in a point's local neighborhood such as roughness, surface orientation, height over ground and others, in very little time. A random forest classifier finally predicts the class-conditional probabilities. The proposed method can process point clouds with many million of points in a matter of minutes.

Online Methods: Vineet et al. (2015) propose an end-to-end system which processes data incrementally and performs real-time dense stereo reconstruction and semantic segmentation of outdoor environments. They achieve this using voxel hashing (Nießner et al. (2013)), a hash-table-driven 3D volumetric representation that ignores unoccupied space in the target environment. Furthermore, they employ an online volumetric mean-field inference technique that incrementally refines the voxel labeling. They are able to achieve semantic reconstruction at real-time rates by harnessing the processing power of modern GPUs.

McCormac et al. (2016) propose a pipeline for dense 3D semantic mapping designed to work online by fusing semantic predictions of a CNN with the geometric information from a SLAM system (ElasticFusion by Whelan et al. (2015)). Specifically, ElasticFusion provides correspondences between 2D frames and a globally consistent map of surfels. Furthermore, they use a Bayesian update scheme which computes the class probabilities for each surfel based on the CNN's predictions. The advantage of using surfel-based surface representations is their ability to fuse long-range information, for instance after a loop-closure has been detected and the poses have been corrected accordingly.

3D CNN: While convolutional networks have proven very successful segmenting 2D images semantically, there exists relatively little work on labeling 3D data using convolutional networks. Huang & You (2016) propose a framework for labeling 3D point cloud data using a 3D Convolutional Neural Network (3D-CNN). Specifically, they compute 3D occupancy grids of size 20^3 centered at a set of randomly generated keypoints. The occupancy and the labels form the input to a 3D CNN, which is composed of convolutional layers, max-pooling layers, a fully connected layer and a logistic regression layer. Due to the dense voxel representation, 3D CNNs are only able to process voxel

grids of very coarse resolution considering the memory limitations of modern GPUs.

To alleviate this problem, Riegler et al. (2017) propose Oct-Nets, a 3D convolutional network, that allows for training deep architectures at significantly higher resolutions. They build on the observation that 3D data (e.g., point clouds, meshes) is often sparse in nature. The proposed OctNet exploits this sparsity property by hierarchically partitioning the 3D space into a set of octrees and applying pooling in a data-adaptive fashion. This leads to a reduction in computational and memory requirements as the convolutional network operations are defined on the structure of these trees and thus can dynamically allocate resources depending on the structure of the input.

6.5. Semantic Segmentation of Street Side Views

One important application of semantic segmentation for autonomous vehicles is to segment street-side images (i.e., building facades) into its components (wall, door, window, vegetation, balcony, store, mailbox etc.). Such semantic segmentations are useful for accurate 3D reconstruction, memory-efficient 3D mapping, robust localization as well as path planning.

Xiao & Quan (2009) propose a multi-view semantic segmentation framework for images captured by a camera mounted on a car driving along the street. Specifically, they define a pairwise MRF across superpixels in multiple views, where the unary terms are based on 2D and 3D features. Furthermore, they minimize color differences for spatial smoothness and use dense correspondences to enforce smoothness across different views. Existing approaches for multi-view semantic segmentation typically require labeling all pixels in all images used for the 3D model which, depending on the semantic segmentation algorithm, can be prohibitively slow. To increase efficiency, Riemenschneider et al. (2014) exploit the inherent redundancy in the labeling of all overlapping images used for the 3D model. They propose an approach that exploits the geometry of a 3D mesh model obtained from multi-view reconstruction to predict the best view for each face of the mesh before performing the actual semantic image labeling. This allows them to accelerate the pipeline by two orders of magnitude.

Gadde et al. (2016b) describe a system for segmentation of 2D images and 3D point clouds of building facades that is fast at inference time and is easily adaptable to new datasets. In contrast to existing methods which exploit the structure of facade images by imposing strong priors, they implement a sequence of boosted decision tree classifiers, that are stacked using auto-context features and learn all correlations from data.

Xiao et al. (2009) propose another method to generate street-side 3D photo-realistic models from images captured at ground level. In particular, they segment each image into semantically meaningful areas, such as building, sky, ground, vegetation or car. Then, they partition buildings into independent blocks and employ a regularization term by exploiting architectural priors in the orthographic view for inference. This allows them to cope with noisy and missing reconstructed 3D data and produces visually compelling results.

Mathias et al. (2016) propose a flexible 3-layered method for segmentation of building facades which avoids the need for
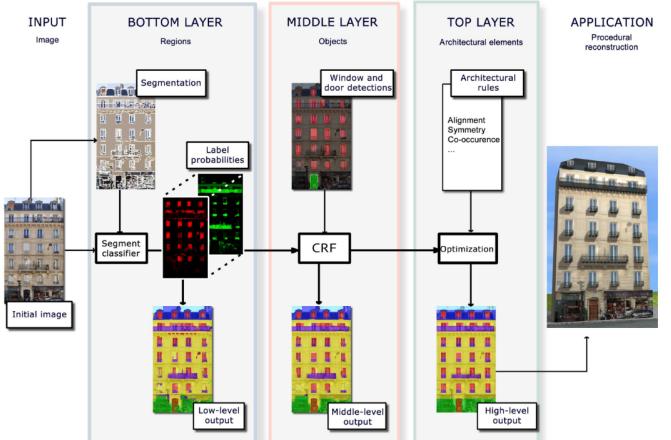

Figure 16: The three-layered approach proposed by Mathias et al. (2016) for facade parsing. They first segment the facade and assign probability distributions to semantic classes considering extracted visual features. In the next layer they use detectors of specific objects such as doors and windows to improve the classifier output from the bottom layer. Finally, they incorporate weak architectural priors and search for the optimal facade labeling using a sampling-based approach. Adapted from Mathias et al. (2016).

explicitly specifying a grammar. First, the facade is segmented into semantic classes which are combined with the output of detectors for architectural elements such as windows and door. Finally, weak architectural priors such as alignment, symmetry, co-occurrence are proposed which encourage the reconstruction to be architecturally consistent. The complete pipeline is illustrated in Figure 16. In contrast to the majority of semantic facade modeling approaches that treat facades as planar surfaces, Martinović et al. (2015) propose an approach for facade modeling which operates directly in 3D. As their approach avoids time consuming conversions between 2D and 3D representations, they obtain substantially shorter runtime. Specifically, they reconstruct a semi-dense 3D point cloud using SfM and classify each point using a Random Forest classifier trained on 3D features. Afterwards, they separate individual facades based on their semantic structure and impose weak architectural priors.

6.6. Semantic Segmentation of Aerial Images

The aim of aerial image parsing is the automated extraction of urban objects from data acquired by airborne sensors. The need for accurate and detailed information for urban objects such as roads is rapidly increasing because of its applications in navigation of autonomous driving systems. For example, aerial image parses can be used to automatically build road maps (even in remote areas) and keep them up-to-date. Furthermore, information from aerial images can be used for localization. However, the problem is challenging because of the heterogeneous appearance of objects like buildings, streets, trees and cars which results in high intra-class variance but low inter-class variance. Furthermore, the complex structure of the prior complicates inference. For instance, roads must form a connected network of thin segments with slowly changing curvatures which meet at junctions. This type of prior knowledge

is more challenging to formalize and integrate into a structured prediction formulation than standard smoothness assumptions.

Wegner et al. (2013) propose a CRF formulation for road labeling in which the prior is represented by cliques that connect sets of superpixels along straight line segments. Specifically, they formulate the constraints as high-order cliques with asymmetric P^N-potentials which express a preference to assign all rather than just some of their constituent superpixels to the road class. This allows the road likelihood to be amplified for thin chains while still being amenable to efficient inference using graph cuts. Wegner et al. (2015) also model the road network using a CRF with long-range, higher-order cliques. However, unlike Wegner et al. (2013), they allow for arbitrarily shaped segments which adapt to more complex road shapes by searching for putative roads with minimum cost paths based on local features. Montoya et al. (2015) extend this formulation to multi-label classification of aerial images with class-specific priors for buildings and roads. In addition to the road network prior of Wegner et al. (2015), they introduce a second higher-order potential for cliques specific to buildings.

In contrast to other methods, Verdie & Lafarge (2014) propose the application of Markov point processes for recovering specific structures from images, including road networks. Markov point processes are a generalization of traditional MRFs which can address object recognition problems by directly manipulating parametric entities such as line segments, whereas MRFs are restricted to labeling problems. Importantly, they implicitly solve the model-selection problem, i.e., they allow for an arbitrary number of variables in the MRF which can be associated with the parameters of the objects of interest. Specifically for road segmentation, the parametric representation of road segments is chosen as a point at the center of mass of the segment and two additional parameters modeling the length and orientation of the road segment.

Aerial Image Parsing using Maps: Instead of framing the problem of detecting topologically correct road network as a semantic segmentation problem, Mattyus et al. (2015) exploit map information from OpenStreetMap (OSM)[27]. OSM is a collection of roads, trails, cafés, railway stations and much more all over the world contributed and maintained by a community of mappers. It provides freely available maps of the road topology in the form of piece-wise linear road segments. Given a road map from OSM, Mattyus et al. (2015) propose an MRF which reasons about the location of the road centerline and its width for each road segment in OSM. In addition, they incorporate smoothness between consecutive line segments by encouraging their widths to be similar. This formulation has the advantage that it enables efficient inference while restricting the road topology to the OSM map.

Fine-grained Image Parsing with Aerial-to-ground Reasoning: While aerial images provide full coverage of a significant portion of the world, they are of much lower resolution than ground images. In aerial imagery the resolution relates to the
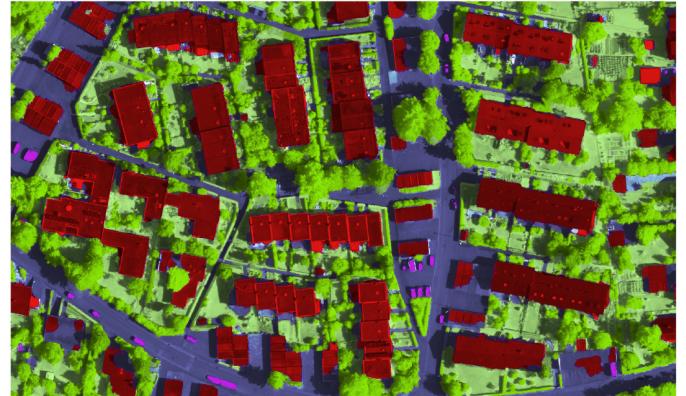

Figure 17: Semantic segmentation of a scene taken from ISPRS Vaihingen using the ensemble of FCNs proposed by Marmanis et al. (2016b). Adapted from Marmanis et al. (2016b).

ground area covered by one pixel. Whereas 1 meter resolution is already a high resolution for satellite imagery, the standard resolution for most image databases (e.g. Google Earth[28]) is 12 inch. Resolutions of 6 to 1 inch are considered high resolutions for aerial imagery and are usually not publicly available. This makes fine grained segmentation from aerial images a challenging problem. On the other hand, ground images provide additional information which enables fine-grained semantic segmentation. Motivated by the complementary nature of these cues, several methods for fine grained segmentation have been recently proposed which jointly reason about co-located aerial and ground image pairs.

Mattyus et al. (2016) extend the approach of Mattyus et al. (2015) by introducing a formulation that reasons about fine-grained road semantics such as lanes and sidewalks. To infer this information, they jointly consider monocular aerial images and high resolution stereo images captured from ground vehicles. Specifically, they formulate the problem as energy minimization in an MRF, inferring the number and location of the lanes for each road segment, all parking spots and sidewalks along with the alignment between the ground and aerial images. Towards this goal, they exploit deep learning to estimate semantics from aerial and ground images and define potentials exploiting both cues. In addition, they define potentials which model road constraints like relationships between parallel roads and the smoothness along roads.

In a related work, Wegner et al. (2016) build a map of trees for urban planning applications from aerial images, street view images and semantic map data. They train CNN based object detection algorithms on human annotated data. Furthermore, they combine the CNN predictions from multiple street view images and aerial images with map data in a CRF formulation to achieve a geolocated fine-grained catalog.

6.6.1. ISPRS Segmentation Challenge

The focus of the ISPRS segmentation challenge[29] (Rottensteiner et al. (2013, 2014)) is detailed 2D semantic segmen-

---

[27]https://www.openstreetmap.org/

[28]https://www.google.com/earth/

[29]http://www2.isprs.org/commissions/comm3/wg4/tests.html

tation of data acquired by airborne sensors as shown in Figure 17. More specifically, the task is to assign labels to multiple urban object categories. The challenge comprises two airborne image datasets, Vaihingen and Potsdam, which have been manually annotated by the six most common land cover classes, namely impervious surfaces, building, vegetation, tree, car, clutter/background. Both areas cover urban scenes. The leaderboards of the datasets Potsdam and Vaihingen are provided in the Table 5. The performance of the approaches is assessed with the F1 scores for the six classes and overall.

Paisitkriangkrai et al. (2015) is one of the best performing methods in the ISPRS segmentation challenge. They propose a semantic pixel labeling method which combines CNN features with hand-crafted features in a pixel-wise CRF formulation to infer a globally consistent labeling that is locally smooth except at edges. Sherrah (2016) propose to use fully-convolutional networks without any downsampling layers to preserve the resolution of the output. In order to make use of elevation data, they propose a hybrid network that combines the pre-trained image features with features based on available digital surface models (DSM) which capture the Earth's surface. Sherrah (2016) achieve the best performance on the ISPRS Potsdam (Table 5a) and competitive results on Vaihingen in Table 5b.

Maggiori et al. (2016) introduce a model which extracts spatial features at multiple resolutions and learns how to combine them in order to integrate local and global information. Audebert et al. (2016) further improved the state-of-the-art for dense scene labeling of aerial images by exploiting the encoder-decoder architecture of SegNet (Badrinarayanan et al. (2015)). In addition, they introduce a multi-kernel convolutional layer for fast aggregation of predictions at multiple scales and perform data fusion from heterogeneous sensors using a residual correction network. Marmanis et al. (2016a) demonstrate the best performance on the ISPRS Vaihingen challenge in Table 5b. They use their previous work Marmanis et al. (2016b) which uses an ensemble of fully convolutional networks to obtain pixelwise classification at full resolution of aerial images. Marmanis et al. (2016a) propose to compensate the loss of spatial resolution due to the pooling layers by combining semantic segmentation with edge detection.

6.7. Road Segmentation

Segmentation of road scenes is a crucial problem in computer vision for applications such as autonomous driving and pedestrian detection. For instance, in order to navigate, an autonomous vehicle needs to determine the drivable free space ahead and determine its own position on the road with respect to the lane markings. However, the problem is challenging due to the presence of a variety of differently shaped objects such as cars and people, different road types and varying illumination and weather conditions.

Munoz et al. (2010) propose an alternative to standard inference in graphical models for semantic labeling of scenes. In particular, they train a sequence of inference models in a hierarchical procedure that captures the context over large regions. This allows them to bypass the difficulties of training structured

prediction models when exact inference is intractable and leads to a very efficient and accurate scene labeling algorithm.

Kuehnl et al. (2012) propose a method that aims to improve appearance-based classification by incorporating the spatial layout of the scene. Specifically, they propose a two-stage approach for road segmentation. First, they represent the road surface and delimiting elements such as curbstones and lane-markings using confidence maps based on local visual features. From these confidence maps, they extract SPatial RAY (SPRAY) features that incorporate global properties of the scene and train a classifier on those features. Their evaluation shows that spatial layout helps especially for the cases where there is a clear structural correspondence between properties at different spatial locations.

Alvarez et al. (2010) propose a Bayesian framework to classify road sequences by combining low-level appearance cues with contextual 3D road cues such as horizon lines, vanishing points, 3D scene layout and 3D road stages. In addition, they extract temporal cues for temporal smoothing of the results. In a follow-up work, Álvarez & López (2011) convert the image into an illuminant invariant feature space to make their method robust to shadows and then apply a classifier to assign a semantic label to each pixel. Mansinghka et al. (2013) propose a inverse-graphics inspired method employing generative probabilistic graphics programs (GPGP) to infer roads in images taken from vehicle mounted cameras. GPGPs consist of a stochastic scene generator for generating random samples from a road scene prior, a graphics renderer for rendering the image segmentation for each sample and a stochastic likelihood model linking the renderer's output and the data.

CNN-based Methods: Almost all existing algorithms for labeling road scenes are based on machine learning where the parameters of the model are estimated from large annotated datasets. To alleviate the burden of annotating large datasets manually, Álvarez et al. (2012) propose a method for road segmentation where noisy training labels for road images are generated using a convolutional neural network trained on a general image database. They further propose a texture descriptor which is based on learning a linear combination of color planes to reduce variability in road texture.

Mohan (2014) propose a scene parsing system using deconvolutional layers in combination with traditional CNNs. Deconvolutional layers learn features that capture mid-level cues such as edge intersections, parallelism and symmetry in image data and thus obtain a more robust representation than regular CNNs. Oliveira et al. (2016) investigate the trade-off between segmentation quality and runtime using U-Nets by Ronneberger et al. (2015). Specifically, they introduce a new mapping between classes and filters at the up-convolutional part of the network to reduce the runtime. They further segment the whole image with a single forward pass, which makes the approach more efficient than patch-based approaches.

To mitigate the difficulties in acquiring human annotations, Laddha et al. (2016) propose a map-supervised deep learning pipeline which does not require human annotations for training a road segmentation algorithm. Instead, they obtain ground

Method	Impervious Surface	Building	Low Vegetation	Tree	Car	Overall
DST – Sherrah (2016)	92.5	96.4	86.7	88	94.7	90.3
UZ – Volpi & Tuia (2016)	89.3	95.4	81.8	80.5	86.5	85.8
SVL – Gerke (2015)	83.5	91.7	72.2	63.2	62.2	77.8

(a) ISPRS Semantic Segmentation (Potsdam)

Method	Impervious Surface	Building	Low Vegetation	Tree	Car	Overall
DLR – Marmanis et al. (2016a)	92.4	95.2	83.9	89.9	81.2	90.3
ONE – Audebert et al. (2016)	91	94.5	84.4	89.9	77.8	89.8
INR – Maggiori et al. (2016)	91.1	94.7	83.4	89.3	71.2	89.5
DST – Sherrah (2016)	90.5	93.7	83.4	89.2	72.6	89.1
ADL – Paisitkriangkrai et al. (2015)	89.5	93.2	82.3	88.2	63.3	88
RIT – Piramanayagam et al. (2016)	90	92.6	81.4	88.4	61.1	88
UOA – Lin et al. (2016b)	89.8	92.1	80.4	88.2	82	87.6
UZ – Volpi & Tuia (2016)	89.2	92.5	81.6	86.9	57.3	87.3
HUST – Quang et al. (2015)	86.9	92	78.3	86.9	29	85.9
ETH_C – Tschannen et al. (2016)	87.2	92	77.5	87.1	54.5	85.9
SVL – Gerke (2015)	86.6	91	77	85	55.6	84.8
UT_Mev – Speldekamp et al. (2015)	84.3	88.7	74.5	82	9.9	81.8

(b) ISPRS Semantic Segmentation (Vaihingen)

Table 5: **ISPRS Semantic Labeling Contest.** Numbers represent F1 scores and overall accuracy.

truth labels based on OpenStreetMap information projected into the image domain using the vehicle pose given by the GPS sensor.

6.7.1. Free Space Estimation

Accurate and reliable estimation of free space and detection of obstacles are core problems that need to be solved to enable autonomous driving. Free space is defined by the available space on the ground surface where navigation of vehicle is guaranteed without collision. Obstacles refer to structures that block the path of the vehicle by sticking out of the ground surface. In contrast to road segmentation approaches, methods for estimating the free-space in front of a vehicle often rely on geometric features as derived from a depth map computed from stereo sensors. However, both approaches can be advantageously combined.

Badino et al. (2007) propose a method for free space estimation by computing stochastic occupancy grids based on stereo information, where cells in a stochastic occupancy grid carry information about the likelihood of occupancy. Stereo information is integrated over time in order to reduce depth uncertainty. The boundary between free space and occupied space is robustly obtained using dynamic programming on the occupancy grid. This work laid the foundations for the Stixel representation, see Section 4 for an in-depth discussion. While the original method of Badino et al. (2007) makes the assumption of a planar road surface, this assumption is often violated in practice. To tackle more complicated road surfaces, Wedel et al. (2009) propose an algorithm which models non-planar road surfaces using B-splines. The surface parameters are estimated from stereo measurements and tracked over time using a Kalman filter.

Suleymanov et al. (2016) propose an online system to detect and drive on collision-free traversable paths, based on stereo es-

timation using a variational approach. In addition to free space detection, their approach also establishes a semantic segmentation of the scene, where labels include ground, sky, obstacles and vegetation. Fisheye cameras provide a wider field of view compared to regular cameras and allow for detection of obstacles closer to the car. Häne et al. (2015) propose a method for obstacle detection using monocular fisheye cameras. In order to reduce runtime, they avoid using visual odometry systems to provide accurate vehicle poses and instead rely on less accurate pose estimates from the wheel odometry.

Long Range Obstacle Detection: The accuracy of obstacle detection methods at long range is a crucial factor for timely obstacle localization when the observer (i.e., the ego-vehicle) moves at high speed. Unfortunately, the error of stereo vision system increases quadratically with depth in contrast to laser range sensors or radar which do not suffer from this problem. To tackle this problem, Pinggera et al. (2015, 2016) propose long range obstacle detection algorithms using stereo vision by exploiting geometric constraints on camera motion and planarity to formulate obstacle detection as a statistical hypothesis testing problem. Specifically, independent hypothesis tests are performed on small local patches distributed across the input images where free-space and obstacles are represented by the null and alternative hypothesis respectively. The detection results for an exemplary scene from their novel dataset is illustrated in Figure 18.

Ground truth annotation

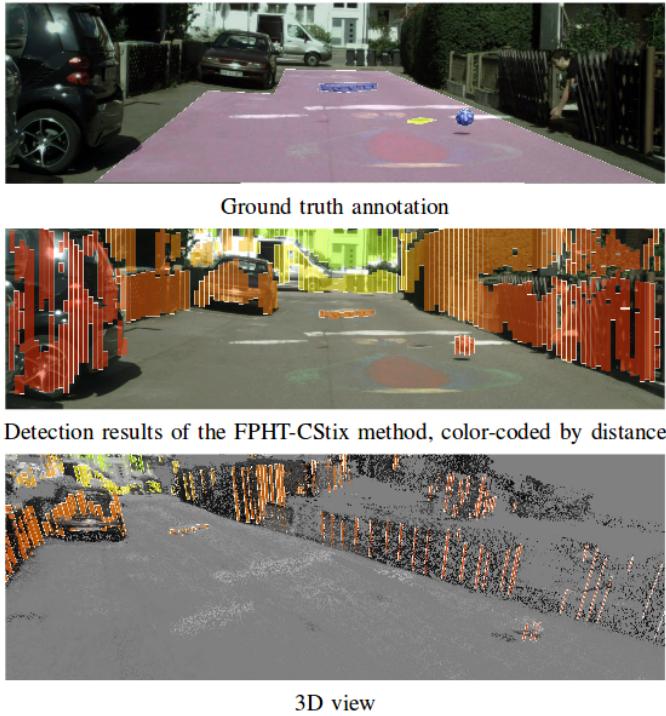Detection results of the FPHT-CStix method, color-coded by distance

3D view

Figure 18: The figure is adapted from Pinggera et al. (2016) and shows the detected obstacles of the proposed approach on the Lost and Found dataset.

7. Reconstruction

7.1. Stereo

Stereo estimation is the process of extracting 3D information from 2D images captured by stereo cameras, without need for special range measurement devices. In particular, stereo algorithms estimate depth information by finding correspondences in two images taken at the same point in time, typically by two cameras mounted next to each other on a fixed rig. These correspondences are projections of the same physical surface in the 3D world. Depth information is crucial for applications in autonomous driving or driver assistance systems. Accurate estimation of dense depth maps is a necessary step for 3D reconstruction, and many other problems such as obstacle detection, free space analysis, and tracking benefit from the availability of depth estimates.

Taxonomies: Multiple taxonomies for stereo matching have been proposed in the literature. Guided by the computational restrictions, the earliest one is based on the density of the output (Franke & Joos (2000)). Feature-based methods provide only sparse depth maps based on edges while area-based methods, such as block matching, generate dense outputs at the expense of computation time. A more recent and commonly referred taxonomy of stereo algorithms is based on the optimization as local and global. Local methods compute the disparity by simply selecting the lowest matching cost which is known as the winner takes all (WTA) solution. Global methods formulate disparity computation as an energy-minimization framework based on the smoothness assumption between neighboring pixels or regions. There are various ways of finding the minimum of a global energy function, including variational approaches in continuous domain and discrete approaches using dynamic programming, Graph Cuts, and Belief Propagation.

Matching Cost Function: Stereo matching is a correspondence problem where the goal is to identify the matching points between left and right image based on a cost function. The algorithms usually assume images are rectified, and the search space is reduced to a horizontal line where the correspondence between a left and right point is encoded by the distance on this line, which is defined as disparity. The matching cost computation is the process of computing a cost function at each pixel for all possible disparities which takes its minimal value at the true disparity. However, it is hard to design such a cost function in practice, therefore stereo algorithms make the assumption of constant appearance between matching points. This assumption is often violated in real-world situations, such as cameras with slightly different settings causing exposure changes, vignetting, image noise, non-Lambertian surfaces, illumination changes, etc. Hirschmüller & Scharstein (2007) call these changes radiometric differences and systematically investigate their effect on commonly used matching cost functions, namely absolute differences, filter-based costs (LoG, Rank and Mean), hierarchical mutual information (HMI), and normalized cross-correlation. They found that the performance of a cost function depends on the stereo method that uses it. On images with simulated and real radiometric differences, rank filter performed best for correlation-based methods. For global methods, in tests with global radiometric changes or noise, HMI performed best, while in the presence of local radiometric variations, Rank and LoG filters performed better than HMI. Qualitative results show that filter-based cost cause blurred object boundaries when used with global methods. None of the matching costs evaluated could succeed at handling strong lighting changes.

SGM: Semi-Global Matching (SGM) (Hirschmüller (2008)) has become very influential due to its speed and high accuracy as evidenced in various benchmarks such as Middlebury (Scharstein & Szeliski (2002)) or KITTI (Geiger et al. (2012b)). SGM is also recently used on top of CNN features, since simply outputting the most likely configuration for every pixel is not competitive with modern stereo algorithms (Žbontar & LeCun (2016); Luo et al. (2016)). The energy function has two levels of penalization for small and large disparity differences with a weighting based on the local intensity gradient for the latter one. The energy is calculated by summing costs along 1D paths from multiple directions towards each pixel using dynamic programming and the result is determined by WTA. There are a couple of follow-up works investigating the practical and theoretical sides of SGM. Gehrig et al. (2009) propose a real-time, low-power implementation of the SGM with algorithmic extensions for automotive applications on a reconfigurable hardware platform. Drory et al. (2014) offer a principled explanation for the success of SGM by clarifying its relation to belief propagation and tree-reweighted message passing with an uncertainty measure as an outcome.

The performance of SGMs can be further improved by incorporating confidences of the stereo estimation. Seki & Polle-

feys (2016) leverage CNNs to predict the confidences for stereo estimations. Taking into account ideas from conventional confidences features, that neighboring pixel which are consistent are more likely to be correct and the disparity estimated from the other image should correspond, they design a two channel disparity patch which is used as input for the CNN. In order to acquire dense disparity, the confidences are incorporated into SGM by weighting each pixel according to the estimated confidence.

Variable Baseline/Resolution: Stereo estimates can be fused to yield a more complete reconstruction of the static parts of the three-dimensional scene. However, assuming fixed baseline, focal length, field of view might not always be the best strategy. Gallup et al. (2008) point out two problems with traditional stereo methods: dropping accuracy in the far range and unnecessary computation time spent in the near range. Given that choice of views for stereo is quite flexible in many applications such as structure from motion, Gallup et al. (2008) propose to dynamically select the best cameras with the appropriate baseline for accurate estimation in the far range from a set of possible cameras recording images at the same time. Further, they reduce the resolution to speed up the computation in the near range. In contrast to traditional fixed-baseline stereo, the proposed variable baseline/resolution stereo algorithm achieves constant accuracy over the reconstructed volume by evenly spreading the computation throughout the volume.

Planarity: The inherent ambiguity in appearance based matching costs can be overcome by regularization, i.e., by introducing prior knowledge about the expected disparity map into the stereo estimation process. The simplest prior favors neighboring pixels to take on the same disparity value. However, such generic smoothness priors fail to reconstruct poorly-textured and slanted surfaces, as they favor fronto-parallel planes. A more generic approach to handle arbitrary smoothness priors is using higher-order connections beyond pairwise. Higher-order priors are able to express more realistic assumptions about depth images, but usually at additional computational cost. One very common way to deal with slanted surfaces in the literature is to assume piecewise planarity. Geiger et al. (2010) build a prior over the disparity space by forming a triangulation on a set of robustly matched correspondences, called support points. This reduces matching ambiguities and results in an efficient algorithm by restricting the search to plausible regions. Gallup et al. (2010) first train a classifier to segment an image into piecewise planar and non-planar regions and then enforce a piecewise planarity prior only for planar regions. Non-planar regions are modeled by the output of a standard multi-view stereo algorithm.

Variational Approaches: Similarly, in variational approaches, commonly used smoothness prior, Total Variation (TV) does not produce convincing results in the presence of weak and ambiguous observations, since it encourages piecewise constant regions leading to stair-casing artifacts. Haene et al. (2012) introduce patch-based priors into a TV framework in the form of small, piecewise planar dictionaries. Total Generalized Varia-
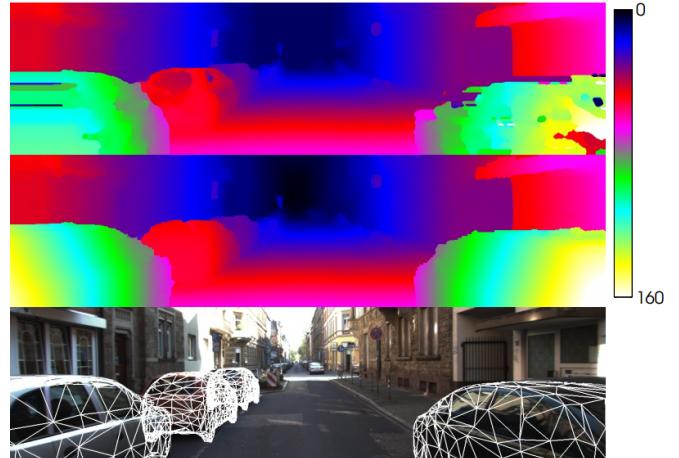

Figure 19: Resolving stereo matching ambiguities using object knowledge. Stereo methods often fail at reflecting, textureless or semi-transparent surfaces (top, Žbontar & LeCun (2016)). By using object knowledge, Güney & Geiger (2015) encourage disparities to agree with plausible surfaces (center). This improves results both quantitatively and qualitatively while simultaneously recovering the 3D geometry of the objects in the scene (bottom). Adapted from Güney & Geiger (2015).

tion (TGV) (Bredies et al. (2010)) is argued to be a better prior than TV, since it does not penalize piecewise affine solutions. However, it is restricted to convex data terms in contrast to TV, where global solutions can be computed even in the presence of non-convex data terms. Coarse-to-fine approaches as an approximation to non-convex problem of stereo matching often end up with loss of details. To preserve fine details, Kuschk & Cremers (2013) integrate an adaptive regularization weight into the TGV framework by using edge detection and report improved results compared to a coarse-to-fine approach. Ranftl et al. (2013) obtain even better results by proposing a decomposition of the non-convex functional into two subproblems which can be solved globally where one is convex, and the other can be made convex by lifting the functional to a higher dimensional space.

State-of-the-art: In Table 6 we show the ranking of stereo methods on the KITTI stereo 2015 benchmark. The KITTI benchmark reports the percentage of erroneous (bad) pixels over background regions (D1-bg), foreground regions (D1-fg) and over all regions (D1-all). The best performing method Güney & Geiger (2015) use object knowledge to compensate for the weak data term on the reflecting and textureless surfaces. Seki & Pollefeys (2016) achieve the best performance on background regions with the prediction of stereo correspondence confidences and integration into SGM. Recently, deep learning approaches (Žbontar & LeCun (2016); Luo et al. (2016); Mayer et al. (2016)) were proposed achieving state-of-the-art performance. The deep learning approach presented by Mayer et al. (2016) is one of the fastest approaches.

Superpixels: An alternative way of modeling piecewise planarity is to explicitly partition the image into superpixels and modeling the surface at each superpixel as a slanted plane (Yamaguchi et al. (2012); Güney & Geiger (2015)). However, care

Method	D1-bg	D1-fg	D1-all	Density	Runtime
Displets v2 – Güney & Geiger (2015)	3.00 %	5.56 %	3.43 %	100.00 %	265 s / 8 cores
PBCP – Seki & Pollefeys (2016)	2.58 %	8.74 %	3.61 %	100.00 %	68 s / GPU
MC-CNN-acrt – Žbontar & LeCun (2016)	2.89 %	8.88 %	3.89 %	100.00 %	67 s / GPU
PRSM – Vogel et al. (2015)	3.02 %	10.52 %	4.27 %	99.99 %	300 s / 1 core
DispNetC – Mayer et al. (2016)	4.32 %	4.41 %	4.34 %	100.00 %	0.06 s / GPU
Content-CNN – Luo et al. (2016)	3.73 %	8.58 %	4.54 %	100.00 %	1 s / GPU
SPS-St – Yamaguchi et al. (2014)	3.84 %	12.67 %	5.31 %	100.00 %	2 s / 1 core
MDP – Li et al. (2016a)	4.19 %	11.25 %	5.36 %	100.00 %	11.4 s / 4 cores
OSF – Menze & Geiger (2015)	4.54 %	12.03 %	5.79 %	100.00 %	50 min / 1 core
CSF – Lv et al. (2016)	4.57 %	13.04 %	5.98 %	99.99 %	80 s / 1 core
MBM – Einecke & Eggert (2014)	4.69 %	13.05 %	6.08 %	100.00 %	0.13 s / 1 core
AABM – Einecke & Eggert (2013)	4.88 %	16.07 %	6.74 %	100.00 %	0.08 s / 1 core
SGM – Hirschmüller (2008)	5.15 %	15.29 %	6.84 %	100.00 %	4.5 min / 1 core
ELAS – Geiger et al. (2010)	7.86 %	19.04 %	9.72 %	92.35 %	0.3 s / 1 core
CostFilter – Rhemann et al. (2011)	17.53 %	22.88 %	18.42 %	100.00 %	4 min / 1 core
OCV-BM – Bradski & Kaehler (2008)	24.29 %	30.13 %	25.27 %	58.54 %	0.1 s / 1 core
VSF – Huguet & Devernay (2007)	27.31 %	21.72 %	26.38 %	100.00 %	125 min / 1 core
MST – Yang & Nevatia (2012)	45.83 %	38.22 %	44.57 %	100.00 %	7 s / 1 core

Table 6: **KITTI 2015 Stereo Leaderboard.** Numbers correspond to percentages of bad pixels according to the 3px/5% criterion defined in Menze & Geiger (2015) in background (bg), foreground (fg) or all regions. The methods below the horizontal line are older entries, serving as reference.

must be taken that the superpixelization is indeed an oversegmentation of the image with respect to planarity, i.e., that no superpixel contains two surfaces which are not co-planar. Yamaguchi et al. (2012) jointly reason about occlusion boundaries and depth in a hybrid MRF composed of both continuous and discrete random variables. Güney & Geiger (2015) use a similar framework to incorporate object-category specific 3D shape proposals which regularize over larger distances. By leveraging semantic segmentation and 3D CAD models, they resolve ambiguities in reflective and textureless regions originating from highly specular surface of cars in the scene as shown in Figure 19.

Deep Learning: In the last years, deep learning approaches (Mayer et al. (2016); Žbontar & LeCun (2016); Luo et al. (2016)) gained popularity in stereo estimation. Mayer et al. (2016) adapt the encoder-decoder architecture proposed by Dosovitskiy et al. (2015) that was used for optical flow estimation (see Section 8.1). The encoder computes abstract features while the decoder reestablishes the original resolution with additional crosslinks between the contracting and expanding network parts. In contrast to the encoder-decoder architecture, Žbontar & LeCun (2016); Luo et al. (2016) use Siamese network which consists of two sub-networks with shared weights and a final score computation layer. The idea is to train the network for computing the matching cost by learning a similarity measure on small image patches. Žbontar & LeCun (2016) define positive/negative examples as matching and non-matching patches and use a margin loss to train either a fast architecture with a simple dot-product layer in the end or a slow but more accurate architecture which learns score computation with a set of fully connected layers. Luo et al. (2016) use a similar architecture, but formulate the problem as multi-class classification over all possible disparities to capture correlations between dif-
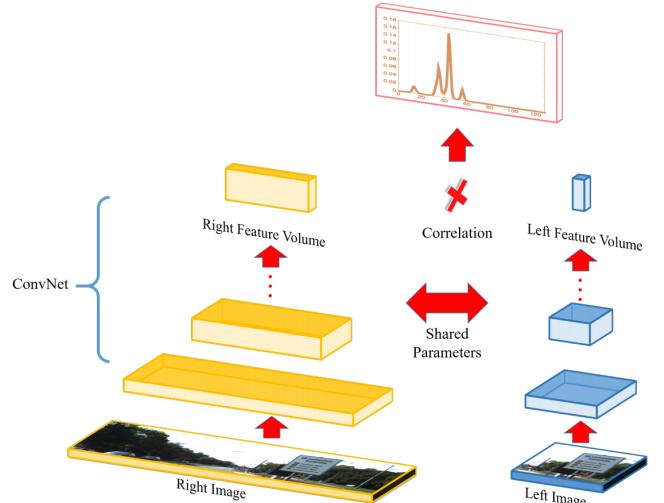

Figure 20: Deep learning for stereo matching. A Siamese network is trained to extract marginal distributions over all possible disparities for each pixel. Adapted from Luo et al. (2016).

ferent disparities implicitly as visualized in Figure 20.

Discussion: Stereo estimation has shown great progress in the last years both in terms of accuracy and efficiency. However, some inherent problems refrain it from being marked as solved. Stereo matching is ultimately searching for correspondences in two images based on the assumption of constant appearance. However, appearance frequently changes by cues different than geometry, furthermore occluded regions or pixels leaving the frame cannot be matched. Therefore, failure in those cases is inevitable for methods that solely rely on appearance matching without any other prior assumptions about the geometry. We

show accumulated errors of top 15 methods on KITTI stereo benchmark Geiger et al. (2012b) in Figure 21. The most common example of failure case in the autonomous driving context are car surfaces due to shiny and reflective regions. Güney & Geiger (2015) specifically address this problem by integrating prior knowledge on possible car shapes. Similarly, windows that are reflective and transparent cannot be matched reliably. As concluded by Hirschmüller & Scharstein (2007), strong illumination changes constitute another common source of error such as inside a tunnel or over-exposure on road surfaces. Pixels leaving the frame and occlusions often cause errors for many methods and both require reasoning beyond matching and local interactions. Other specific examples of problematic regions include thin structures like traffic signs, or repetitive ones like fences.

7.2. Multi-view 3D Reconstruction

The goal of multi-view 3D reconstruction is to model the underlying 3D geometry by inverting the image formation process often under certain prior or smoothness assumptions. In contrast to two-view stereo, multi-view reconstruction algorithms in particular address the problems of varying viewpoints and the complete reconstruction of 3D scenes from more than two and potentially a very large number of images. If the camera parameters are known, solving for the 3D geometry of the scene is equivalent to solving the correspondence problem, based on a photo-consistency function which measures the agreement between different viewpoints.

Taxonomies: Several categorizations of multi-view reconstruction algorithms have been proposed in the literature, typically considering the form of the photo-consistency function, the scene representation, visibility computation, priors, and initialization requirements as in Seitz et al. (2006). From an application perspective, the scene representation is a common way of classifying multi-view reconstruction approaches into depth map, point cloud, mesh, and volumetric.

Representations: Depth Map: The depth map representation typically consists of a depth map for each input view estimated with a 3D modeling pipeline which starts with image matching followed by pose estimation and dense stereo. This representation is usually preferred in scene analysis due to its flexibility and scalability to large scenes. One strategy which is particularly effective for urban scenes is Plane Sweeping Stereo algorithm (Collins (1996)). It sweeps a family of parallel planes in a scene, projects images onto a plane via planar homographies, then evaluates photo-consistency values on each plane. In large scenes, one of the challenges is to handle massive amount of data in real-time. Pollefeys (2008) propose a large scale, real-time 3D reconstruction system based on depth map representation. The real-time performance is achieved by incorporating a set of components which are particularly efficient on typical urban scenes such as a 2D feature tracker with automatic gain adaptation for handling large dynamic range in natural scenes, and parallel implementations of plane sweeping stereo and depth map fusion on GPU.

Representations: Point-cloud: In contrast to a partial depth map for each view, point-cloud or patch based surface representations reconstruct a single 3D point-cloud model using all the input images. Under spatial consistency assumptions, the point-cloud on the surface of the scene can grow or expand which provides easy model manipulation such as merging and splitting. The representative work for these kind of approaches is Patch-based Multi-View Stereo (PMVS) by Furukawa & Ponce (2010). PMVS starts with a feature matching step to generate a sparse set of patches and then iterate between a greedy expansion step and a filtering step to make patches dense and remove erroneous matches.

Representations: Volumetric: Volumetric approaches represent geometry on a regularly sampled 3D grid, i.e. volume, either as a discrete occupancy function (Kutulakos & Seitz (2000)) or a function encoding distance to the closest surface (level-set) (Faugeras & Keriven (1998)). More recent approaches use a probability map defined at regular voxel locations to encode the probability of occupancy (Bhotika et al. (2002); Pollard & Mundy (2007); Ulusoy et al. (2015)). The amount of memory required is the main limitation for volumetric approaches. There is a variety of methods for dealing with this problem such as voxel hashing (Nießner et al. (2013)) or a data adaptive discretization of the space in the form of a Delaunay triangulation (Labatut et al. (2007)). One effective solution is an octree data structure which is essentially an adaptive voxel grid to allocate high resolution cells only near the surfaces.

Representations: Mesh or Surface: The final representation in reconstruction is typically triangular mesh-based surfaces. Volumetric surface extraction fuses 3D information from an intermediate representation such as depth maps, point clouds, volumes or scans into a single, clean mesh model. Seminal work by Curless & Levoy (1996) proposes an algorithm to accumulate surface evidence into a voxel grid using signed distance functions. The surface is implicitly represented as the zero crossing of the aggregated signed distance functions. It can be extracted using the Marching Cube algorithm Lorensen & Cline (1987) or using volumetric graph cuts to label each voxel as interior or exterior. There are approaches which directly start from images and refine a mesh model using an energy function composed of a data term based on photo-consistency function and a regularization term for smoothness. In these approaches, the energy is usually optimized using gradient descent, where the movement of each vertex is determined by the gradient of the objective function.

Urban Reconstruction: In this survey, we focus on multi-view reconstruction from an autonomous driving perspective which mainly concerns the reconstruction of large urban areas, up to whole cities. The goal of urban reconstruction algorithms is to produce fully automatic, high-quality, dense reconstructions of urban areas by addressing inherent challenges such as lighting conditions, occlusions, appearance changes, high-resolution inputs, and large scale outputs. Musialski et al. (2013) provide a survey of urban reconstruction approaches by following an output-based ordering, namely buildings and se-

Figure 21: **KITTI 2015 Stereo Analysis.** Accumulated errors of 15 best-performing stereo methods published on the KITTI 2015 Stereo benchmark. Red colors correspond to regions where the majority of methods results in bad pixels according to the 3px/5% criterion defined in Menze & Geiger (2015). Yellow colors correspond to regions where some of the methods fail. Regions which are correctly estimated by all methods are shown transparent.

33

mantics, facades and images, and finally blocks and cities.

Input Data: Musialski et al. (2013) point out that ground, aerial and satellite imagery, as well as Light Detection and Ranging (LiDAR) scans are the most commonly used sensors for urban reconstruction. Ground-level imagery is the most prevalent one due to easy acquisition, storage and exchange. Aerial and satellite imagery have become more easily available due to the advances of Web-mapping projects. In contrast to aerial or multi-view imagery, satellite imagery provides a worldwide coverage at a high frequency with lower costs, but also with lower resolution. LiDAR delivers semi-dense 3D point-clouds which are fairly precise, both ground-level and aerial. Some approaches also incorporate several of these data types together in order to combine their complementary strengths. To deal with the challenging conditions of outdoor scenes, other methods leverage additional data sources, like Digital Surface Models (DSMs) which capture the Earth's surface. DSMs are 2.5D representations of an urban scene that provide a height for each point on a regular grid. In the following, we provide recent examples of different input modalities.

Stereo Sequences: Cornelis et al. (2008) point out that the extraction of detailed 3D information from video streams incur high computational cost for reconstruction algorithms. By keeping the necessary level of detail low, they focus on creating compact, memory efficient 3D city models from a stereo pair, at high speed based on simplified geometry assumptions, namely ruled surfaces for facade and road surfaces. Since objects such as cars which are prevalent in urban scenes violate these assumptions, they integrate the detection and localization of cars into the reconstruction. By leveraging efficient stereo matching, Geiger et al. (2011) propose a system to generate accurate 3D reconstructions of static scenes from stereo sequences in real-time. For online reconstruction, they employ two threads: the first thread performs feature matching and ego-motion estimation, while the second thread performs dense stereo matching and 3D reconstruction.

Digital Surface Models (DSM): Digital Surface Models are either generated from aerial LiDAR point clouds or Multi-View Stereo (MVS) and adapted to geometric descriptions of urban scenes. MVS-based DSMs can be very noisy and therefore Lafarge et al. (2010) propose to generate DSMs from MVS imagery by reconstructing buildings with an assemble of simple urban structures extracted from a library of 3D parametric blocks. In contrast to MVS-based DSMs, laser scans have been also very popular to acquire 3D city models. Lafarge & Mallet (2012) provide a more complete description of urban scenes by simultaneously reconstructing trees and topologically complex ground surfaces in addition to the buildings from point clouds generated by aerial data. They model the original hybrid representation of buildings by combining two different types of 3D representations: primitives for regular parts of buildings as in Lafarge et al. (2010) and mesh patches for modeling atypical surfaces such as irregular roofs.

Air- and Street-level: Früh et al. (2005) register a series of vertical 2D surface scans and camera images to airborne data

(DSMs) to generate textured facade meshes of cities. They propose a class of data processing techniques to create visually appealing facade meshes by removing noisy foreground objects and filling holes in the geometry and texture of building facades. Bódis-Szomorú et al. (2016) point out that airborne and mobile mapping data provide complementary information and need to be exploited together in order to produce complete and detailed large-scale city models. Airborne sensors can acquire roof structures, ground, and vegetation at large scale while on-road mobile mapping by multi-view stereo approaches or LiDAR provide the facade and street-side details. They propose a solution to fuse a detailed on-road mobile mapping and a coarser but more complete point cloud from airborne acquisition in a joint surface mesh. Their evaluation shows that the quality of the model improves substantially by fusing street-side details into the airborne model.

Stereo Satellite: Duan & Lafarge (2016) propose a method to produce compact 3D city models composed of ground and building objects from stereo pairs of satellite images. They represent the scene using convex polygons and perform joint classification and reconstruction of the semantic class (ground, roof, and facade) and the elevation of each polygon. Although their evaluation shows that the obtained results are not as accurate as LiDAR scans, the proposed method can produce fast, compact, and semantic-aware models robust to low resolution and occlusion problems.

7.3. Reconstruction and Recognition

In autonomous driving, it is important to understand both the structural and semantic information of the surroundings. Traditionally, image segmentation methods employ priors entirely in the 2D image domain, i.e., spatial smoothness terms, and reconstruction methods usually encourage piecewise smooth surfaces. It has been long argued that semantics and 3D reconstruction carry valuable information to each other. Similarly to stereo, the motivation to incorporate semantics in reconstruction is photo-consistency failing in case of imperfect and ambiguous image information due to specularities, lack of texture, repetitive structures, or strong lighting changes. Semantic labels provide geometric cues about likely surface orientations at a certain location and help resolving inherent ambiguities. 3D reconstruction lifts the reasoning from 2D to 3D and acts as a strong regularizer by enforcing geometric consistency over multiple images for segmentation.

Planarity and Primitives: Micusik & Kosecka (2009) present a method to overcome these difficulties by exploiting image segmentation cues as well as presence of dominant scene orientations and piecewise planar structures. In particular, they adopt a super-pixel based dense stereo reconstruction method by using the Manhattan world assumption with three orthogonal plane normals in the MRF formulation. Another way of exploiting piecewise planar structures and the shape repetition is to use primitives such as planes, spheres, cylinders, cones and tori (Lafarge et al. (2010); Lafarge & Mallet (2012); Lafarge et al. (2013)). Primitive arrangement-based approaches provide compactness and reduce complexity. However, they
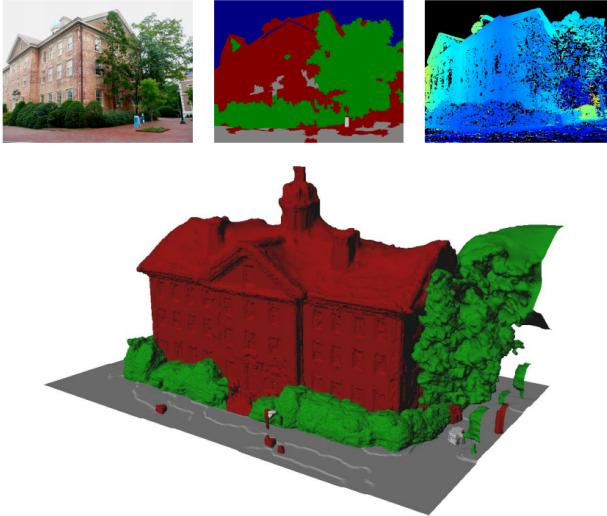
Figure 22: Joint 3D scene reconstruction and class segmentation by Haene et al. (2013). The top row shows an example of the input images and its corresponding 2D semantic segmentation and depth map. The result of joint optimization over class segmentation and geometry is shown at the bottom. Adapted from Haene et al. (2013).

remain simplistic representations and fail to model fine details and irregular shapes. Therefore, Lafarge et al. (2013) propose a hybrid approach which is both compact and detailed. Starting from an initial mesh-based reconstruction, they use primitives for regular structures such as columns and walls, while irregular elements are still described by meshes for preserving details.

Volumetric: Volumetric scene reconstruction typically segments the volume into occupied and free-space regions. Haene et al. (2013) present the mathematical framework to extend it to a multi-label volumetric segmentation framework which assigns object classes or a free-space label to voxels as shown in Figure 22. They first learn appearance likelihoods and class-specific geometry priors for surface orientations from the training data. Then, these data-driven priors are used to define unary and pairwise potentials in a continuous formulation for volumetric segmentation. Joint reasoning benefits from typical class-specific geometry, such as the normals of the ground plane pointing upwards. In addition, it provides a class-specific smoothness prior in cases of weak cues for the scene geometry. Their evaluation shows the benefit of such a prior over standard smoothness assumptions such as Total Variation.

Zhou et al. (2015) propose a method for 3D reconstruction of street scenes from a sequence of fisheye cameras by introducing semantic priors. Motivated by recurring objects of similar 3D shapes in outdoor scenes, they first localize buildings and vehicles using 3D object detectors and then jointly reconstruct them while learning a volumetric model of their shape. This allows to reduce noise while completing missing surfaces as objects of similar shape benefit from all observations of the respective category.

Monocular Video: Failures in multi-view stereo cause problems for approaches like Haene et al. (2013) which require dense

depth measurements. Using a monocular image stream as input, Kundu et al. (2014) propose another joint reasoning approach over a sparse point cloud from SfM and dense semantic labeling of the frames. This way, 3D semantic representation is temporally coherent without additional cost. They model the problem with a higher order CRF in 3D which allows realistic scene constraints and priors such as 3D object support. In addition, they explicitly model the free space which provides cues to reduce ambiguities, especially along weakly supported surfaces. Their evaluation on monocular datasets Camvid and Leuven shows improved 3D structure compared to traditional SfM and state-of-the-art multi-view stereo as well as better segmentation quality over video segmentation methods in terms of both per pixel accuracy and temporal consistency.

Volumetric: Large-scale: Previous works on semantic reconstruction (Haene et al. (2013); Kundu et al. (2014)) are limited to small scenes and low resolution, because of their large memory footprint and computational cost. To scale them up to large scenes, Blaha et al. (2016) point out that high resolution is not required for large regions such as free space, parts under the ground, or inside the building. They propose an extension of Haene et al. (2013) by employing an adaptive octree data structure with coarse-to-fine optimization, in an application to generate 3D city models from terrestrial and aerial images. Starting from a coarse voxel grid, they solve a sequence of problems in which the solution is gradually refined only near the predicted surfaces. The adaptive refinement saves memory and runs much faster while still being as accurate as the fixed voxel discretization at the highest target resolution, both in geometric reconstruction and semantic labeling.

Besides the spatial extent, the number of different semantic labels is also a problem for scalability due to increasing memory requirements. The complexity is quadratic in the number of labels due to indicator variables for the transitions between the different labels. Cherabier et al. (2016) propose to divide the scene into blocks in which only a set of relevant labels is active, since absence of many semantic classes from a specific block can be determined early on. Accordingly, they can deactivate a label right from the beginning of the optimization which leads to a more efficient processing. The set of active labels in each block is updated during the iterative optimization to recover from wrong initializations. Their evaluation shows that they can increase the number of labels from six to nine with a significant gain in memory compared to Haene et al. (2013).

Shape Priors: Advances in sensors to acquire 3D shapes and the performance of object detection algorithms have encouraged the use of 3D shape priors in 3D reconstruction. Dimensionality reduction is an effective and popular way of representing shape knowledge. Early approaches use linear dimensionality reduction such as PCA to capture the shape variance in low dimensional latent shape spaces. More recent approaches use nonlinear dimensionality reduction such as Gaussian Process Latent Variable Models (GP-LVM) (Dame et al. (2013)).

Dame et al. (2013) investigate the importance of shape priors in a monocular SLAM approach. In parallel with depth estimation, they refine an object's pose, shape and scale to match an

initial segmentation and depth cues. It is finally fused into the volumetric representation. Their experiments show improvement in transparent and specular surfaces, and even in unobserved parts of the scene. In addition to mean shape, Bao et al. (2013) propose to learn a set of anchor points as representative of object shape across several instances. They first perform an initial alignment using 2D object detectors. Next, they align the point cloud from SfM with the mean shape by matching anchor points, and then warp and refine it to approach the actual shape. Their evaluation demonstrates that the model is general enough to learn semantic priors for different object categories such as car, fruit, and keyboard by handling large shape variations across instances.

While previous approaches (Dame et al. (2013); Bao et al. (2013)) try to fit a parametric shape model to input data, Haene et al. (2014) model the local distribution of normals for an object. They propose an object class specific shape prior in the form of spatially varying anisotropic smoothness terms. Similar to multi-label segmentation approach of Haene et al. (2013), they divide the reconstruction into object region and the supporting ground and apply the shape prior only on object to guide the optimization to the right shape.

Data-Driven: Instead of modeling a semantic prior for each object explicitly, Wei et al. (2014) propose a data-driven regularization to transfer the shape information of the disparity or flow from semantically matched patches in the training database using the SIFT flow algorithm. They represent the shape information as the relative relationship of scene properties instead of absolute values. It is mainly for reusability of scene properties, such as modeling disparity of car independent of its position. They compare their data-driven prior against popular smoothness terms on Sintel and show improved performance while being comparable to state-of-the-art on KITTI.

8. Motion & Pose Estimation

8.1. 2D Motion Estimation – Optical Flow

Optical flow is defined as the two dimensional motion of brightness patterns between two images. This definition only represents motion of intensities in the image plane but not the 3D motion of the objects in the scene. Recovering the 3D motion itself is the goal in Scene Flow discussed in Section 8.2. Figure 23 shows the synthetic Yosemite sequence with the optical flow ground truth generated by texture mapping aerial images of Yosemite valley on depth maps of the valley. Optical flow provides important information about the scene and serves as input for several tasks such as ego-motion estimation (Section 8.3), structure-from-motion and tracking (Section 9). The research on this problem started several decades ago with the variational formulation by Horn & Schunck (1981) assuming the brightness of a pixel to be constant over time. Optical flow is an inverse problem in which insufficient information is given to fully specify the solution. The brightness at a pixel provides only one constraint while the unknown motion vector has two components. This is known as the aperture problem and can only be solved by introducing an additional constraint which is
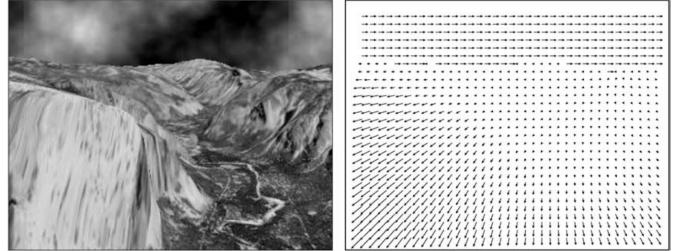

Figure 23: The Yosemite sequence generated by Quam (1984) and the corresponding ground truth flow created by Heeger (1988). The sequence was later incorporated into the Middlebury dataset of Baker et al. (2011). Adapted from Heeger (1988).

usually a smoothness assumption encouraging similar motion vectors between neighboring pixel. Despite the long history of the optical flow problem, occlusions, large displacement and fine details are still challenging for modern methods. A fundamental problem with the optical flow definition is that besides the actual motion of interest, illumination changes, reflections and transparency can also cause intensity changes besides the motion.

Variational Formulation: Traditionally, the optical flow problem has been approached with a variational formulation. Variational methods minimize an energy consisting of a data term, assuming little appearance change over time, and a smoothness term, encouraging similarity between spatial neighbors. Horn & Schunck (1981) introduced the brightness constancy assumption which models the intensity value of a pixel as constant over time. Considering one pixel this assumption yields one equation with two unknowns that cannot be solved as such (aperture problem). To estimate the optical flow an additional constraint is necessary. A common way of regularizing variational optical flow estimation is to encourage similarity of spatially neighboring flow vectors. This prior is motivated by the fact that flow fields are often smooth and discontinuities typically occur only at object boundaries. The original formulation by Horn & Schunck (1981) uses a quadratic penalty function in the data and smoothness term. This has the major limitation that violations of the brightness constancy assumption, like varying illumination conditions, can not be handled. One very popular way to alleviate this problem is using a robust penalty function as proposed by Black & Anandan (1993). In addition, several different data terms have been proposed that are less affected by illumination changes. Vogel et al. (2013) systematically evaluates pixel- and patch-based data costs in a unified testbed on the KITTI dataset (Geiger et al. (2012b)). On real data, they found patch-based terms to perform better than pixel-based terms. Another limitation of the original formulation by Horn & Schunck (1981) is that the homogeneous non-robust smoothness term does not allow flow discontinuities. However, in real world scenes different objects often cause optical flow discontinuities at their boundaries thus violating this assumption. Total Variation regularization used in Zach et al. (2007) replaces the quadratic penalization by the L_1 norm to preserve discontinuities in the flow field. A remaining disadvantage of this model
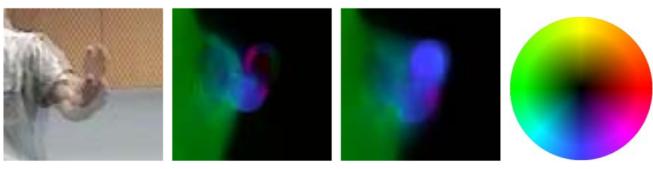
Figure 24: Fast hand motion (left) is an example where classical warping methods fail (center left) but sparse matches introduced by Brox & Malik (2011) help to estimate the flow (center right). The color encoding of the flow is visualized in the right image. Adapted from Brox & Malik (2011).

Figure 25: Trade-off between performance and speed on KITTI 2012 Geiger et al. (2012b). Adapted from Wulff & Black (2015).

is that it favors fronto-parallel surfaces which is not a realistic assumption for real-world scenes. Thus, higher-order regularizations like the Total Generalized Variation (TGV) model have been proposed by Bredies et al. (2010). TGV priors can better represent real data as they leverage a piecewise affine motion model. The non-local Total Generalized Variation by Ranftl et al. (2014) is an extension of this model which enforces the piecewise affine assumption in a local neighborhood. They observed that considering only direct neighbors leads to a decrease of performance in regions where the data term is ambiguous. Zimmer et al. (2011) provide a detailed assessment of image- and flow-driven regularizers for the variational formulation and discuss the qualities of different data terms. Besides the model specifications, the choice of the optimization method and its implementation are additional factors which influence the performance of variational optical flow estimation algorithms. A detailed study of optical flow methods is provided by Sun et al. (2014). They uncover the reasons for the success of modern optical flow methods and propose an approach optimizing a classical formulation with modern techniques.

Sparse Matches: One major challenge, in particular for variational methods, is the estimation of large displacements since usually linear approximations are used that only hold in case of pixel motion. This problem is typically addressed with a coarse-to-fine strategy, estimating the flow on a coarser resolution to initialize the estimation on a finer resolution. While this strategy works for large structures of little complexity, fine geometric details are often lost in the process. Besides, textural details important for correspondence estimation are lost at coarse resolutions, hence leading the optimizer to a local minimum. One example for the loss of fine details is illustrated with a fast moving hand in Figure 24. These problems can be alleviated by integrating sparse features into the variational formulation as proposed by Brox & Malik (2011). The feature matches, obtained from nearest neighbor search on a coarse grid, are used as soft constraint in a coarse-to-fine optimization. While in Figure 24 the warping methods fail to recover the optical flow for the hand, the feature matches lead the optimization to the right solution. Another possibility to deal with large displacements is suggested by Revaud et al. (2015). They replace the coarse-to-fine strategy with an interpolation of sparse matches to initialize a dense optimization at full resolution. Sparse matches are obtained using DeepMatching, a deep neural network matching approach introduced by Weinzaepfel et al. (2013). In con-
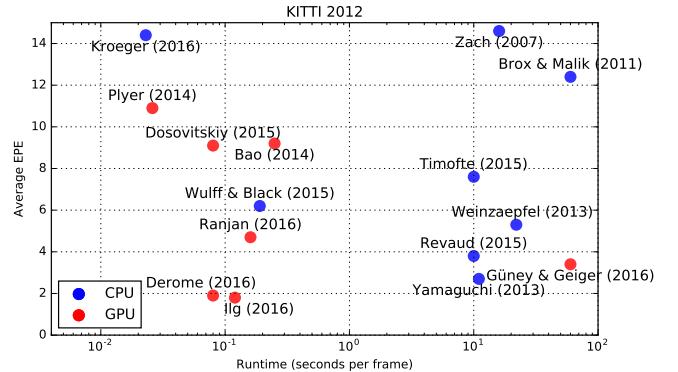
trast to DeepMatching, Menze et al. (2015a) use approximate nearest neighbor search to generate a set of proposals as candidates to be used in a discrete optimization framework. Inference is made feasible by restricting the number of matches to the most likely ones with non-maxima suppression and exploiting the truncated form of the pairwise potentials. Motivated by the success of Siamese networks in stereo (Žbontar & LeCun (2016)) (see Section 7.1), Güney & Geiger (2016) extend this work to learning features for 2D patch matching. They further investigate the importance of the receptive field size exploiting dilated convolutions as proposed by Yu & Koltun (2016) for semantic segmentation. Chen & Koltun (2016) argue that the heuristic pruning used to make inference feasible destroys the highly regular structure of the space of mappings and propose a discrete optimization over the full space. Min-convolutions are used to reduce the complexity and to effectively optimize the large label space using a modified version of Tree-Reweighted Message Passing by Kolmogorov (2006). Wulff & Black (2015) present a different approach to obtain dense optical flow from sparse matches. In their approach, the optical flow field is represented as a weighted sum of basis flow fields learned from reference flow fields which have been estimated from Hollywood movies. They estimate the optical flow by finding the weights which minimize the error with respect to the detected sparse feature correspondences. While this results in overly smooth flow fields, the approach is very fast. Besides, a slower layered approach has been approached which better handles flow discontinuities.

High Speed Flow: With some exceptions (Wulff & Black (2015); Timofte & Gool (2015); Weinzaepfel et al. (2013); Farneback (2003); Zach et al. (2007)) most of the optical flow approaches are very inefficient and can not be applied in real-time which is necessary for applications in autonomous driving. The trade-off between accuracy and speed for different algorithms on KITTI 2012 benchmark Geiger et al. (2012b) is illustrated in Figure 25. The methods based on variational inference yield the best accuracy, however belong to the slowest set of methods for motion estimation. However, the duality based approach for total variation optical flow proposed by Zach et al. (2007) allows an efficient GPU implementation that performs in real-time (30 Hz)

on a resolution of 320×240. Sparse matching approaches are usually more efficient than variational formulations but often need variational refinement as post processing step to achieve subpixel precision. The recent introduction of deep learning to the optical flow problem yielded several almost real-time approaches (Dosovitskiy et al. (2015); Ranjan & Black (2016)) including Ilg et al. (2016) which achieves state-of-the-art performance on popular datasets. These methods will be discussed below. The approach proposed by Kroeger et al. (2016) allows to trade-off accuracy and computational time. They realize fast patch correspondences with inverse search and obtain a dense flow field with the aggregation of patches along multiple scales. This allows them to estimate optical flow with up to 600 Hz at the cost of accuracy.

State-of-the-art: Currently, Sintel Butler et al. (2012) and KITTI Geiger et al. (2012b, 2013) discussed in Section 2 are the most popular datasets for the evaluation of optical flow algorithms. However, in this survey we focus on the autonomous driving application. Therefore, we will only refer to the KITTI leaderboard when we compare methods. Still, optical flow approaches not specifically designed for autonomous driving have a similar ranking on Sintel. In Table 7 we show the leaderboard for the KITTI 2015 benchmark. The performance of methods is assessed using the percentage of outliers, which are flow vectors with the absolute endpoint error (EPE) exceeding 3 pixel and 5% of its true values. The percentage of outliers is averaged over background (Fl-bg), foreground (Fl-fg) and all regions (Fl-all). In addition, the density of the output flow field and the runtime are provided. The best performing methods either learn optical flow end-to-end (Ilg et al. (2016)) or use semantic segmentation to split the scene into independently moving objects Bai et al. (2016); Sevilla-Lara et al. (2016). The best performing approach FlowNet2 (Ilg et al. (2016)) trains a deep neural network to solve the optical flow problem.

Epipolar Flow: In the context of autonomous driving, simplifying assumptions can be used to alleviate the optical flow problem. The assumption of a static scene or the decomposition of a scene into rigidly moving objects allow to treat optical flow as matching problem along epipolar lines radiated from the focus of expansion. Yamaguchi et al. (2013) propose a slanted-plane Markov random field that represents the epipolar flow of each segment with slanted planes. This formulation needs a time consuming optimization and can be avoided with the joint stereo and flow formulation of Yamaguchi et al. (2014). They assume the scene to be static and present a new semi global block matching algorithm using the joint evidence of stereo and video. This formulation allows them to rank third in KITTI 2012 while being 10 times faster than the best performing method. In contrast to these approaches, Bai et al. (2016) use the slanted plane model only for background flow estimation. An instance-segmentation allows them to formulate an independent epipolar flow estimation problem for each moving object. While for KITTI 2012 the advantage of this formulation is not evident because of the static scene, on KITTI 2015 which comprises dynamic scenes they achieve better results (Table 7).

Semantic Segmentation: Scenes in the context of autonomous driving are usually composed of a static background and dynamic moving traffic participants. This observation can be exploited by splitting the scene into independently moving objects. As mentioned above, Bai et al. (2016) extract traffic participants using instance-level segmentation and estimate the optical flow independently for different instances. Sevilla-Lara et al. (2016) use semantic segmentation for optical flow estimation in several ways: on one hand, semantics provide information on object boundaries as well as spatial relationships between objects that are used to reason about depth ordering. On the other hand, the division of the scene allows Sevilla-Lara et al. (2016) to exploit different motion models according to the respective object type, similar to Bai et al. (2016). The motion of planar regions is modeled with homographies, whereas independently moving objects are modeled by affine motions allowing for deviations. Complex objects like vegetation are modeled with a classical spatially varying dense flow field. Finally, the constancy of object identities over time is used to encourage temporal consistency of the optical flow.

Confidences: Considering the remaining challenges in optical flow, a confidence measure to assess the quality of the estimated flow is desirable. Several measures based on spatial and temporal gradients have been proposed (Uras et al. (1988); Anandan (1989); Simoncelli et al. (1991)) that quantify the difficulty to estimate flow for a specific image. In contrast, algorithm-specific measures (Bruhn & Weickert (2006); Kybic & Nieuwenhuis (2011)) have been proposed which give a confidence for an estimation only for a specific group of methods. Learning-based measures like Kondermann et al. (2007, 2008) learn a model that relates flow algorithm success to spatio-temporal image data or the computed flow field. A detailed evaluation of different confidence measures is given by Mac Aodha et al. (2013). In addition, they present another learning based approach which uses multiple feature types such as temporal, texture, distance from images edges, and others, to estimate confidences for the success of a given method.

Deep Learning: Most optical flow approaches do not incorporate any high-level information like semantics which makes it hard to resolve ambiguities. The knowledge about objects and their material property can be used to model reflectance and transparency which would allow to be unaffected by these phenomena. The recent success of convolutional neural networks to learn high-level information have led to the attempt of using them for the optical flow problem. Dosovitskiy et al. (2015) presented FlowNet to learn optical flow end-to-end using a CNN. FlowNet consists of a contracting part which extracts important features and an expanding part which produces the high resolution flow. They propose two different architectures: a simple network stacking the images and a complex network correlating features of the separately processed images. One problem in learning optical flow is the limited amount of training data. KITTI 2012 Geiger et al. (2012b) and KITTI 2015 Menze & Geiger (2015) only provide around 200 training examples each while Sintel Butler et al. (2012) has 1041 training image pairs. Since these datasets are too

Method	Fl-bg	Fl-fg	Fl-all	Density	Runtime
FlowNet2 – Ilg et al. (2016)	10.75 %	15.14 %	11.48 %	100.00 %	0.12 s / GPU
SDF – Bai et al. (2016)	8.61 %	26.69 %	11.62 %	100.00 %	TBA / 1 core
SOF – Sevilla-Lara et al. (2016)	14.63 %	27.73 %	16.81 %	100.00 %	6 min / 1 core
JFS – Hur & Roth (2016)	15.90 %	22.92 %	17.07 %	100.00 %	13 min / 1 core
PCOF-LDOF – Derome et al. (2016)	14.34 %	41.30 %	18.83 %	100.00 %	50 s / 1 core
PatchBatch – Gadot & Wolf (2016)	19.98 %	30.24 %	21.69 %	100.00 %	50 s / GPU
DDF – Güney & Geiger (2016)	20.36 %	29.69 %	21.92 %	100.00 %	1 min / GPU
DiscreteFlow – Menze et al. (2015a)	21.53 %	26.68 %	22.38 %	100.00 %	3 min / 1 core
PCOF + ACTF – Derome et al. (2016)	14.89 %	62.42 %	22.80 %	100.00 %	0.08 s / GPU
CPM-Flow – Hu et al. (2016)	22.32 %	27.79 %	23.23 %	100.00 %	4.2 s / 1 core
MotionSLIC – Yamaguchi et al. (2013)	14.86 %	66.21 %	23.40 %	100.00 %	30 s / 4 cores
FullFlow – Chen & Koltun (2016)	23.09 %	30.11 %	24.26 %	100.00 %	4 min / 4 cores
SPM-BP – Li et al. (2015)	24.06 %	29.56 %	24.97 %	100.00 %	10 s / 2 cores
EpicFlow – Revaud et al. (2015)	25.81 %	33.56 %	27.10 %	100.00 %	15 s / 1 core
DeepFlow – Weinzaepfel et al. (2013)	27.96 %	35.28 %	29.18 %	100.00 %	17 s / 1 core
DWBSF – Richardt et al. (2016)	40.74 %	35.53 %	39.87 %	100.00 %	7 min / 4 cores
LDOF – Brox & Malik (2011)	40.81 %	35.42 %	39.91 %	95.89 %	86 s / 1 core
HS – Sun et al. (2014)	39.90 %	53.59 %	42.18 %	100.00 %	2.6 min / 1 core
DB-TV-L1 – Zach et al. (2007)	47.52 %	50.23 %	47.97 %	100.00 %	16 s / 1 core
HAOF – Brox et al. (2004)	49.89 %	52.28 %	50.29 %	100.00 %	16.2 s / 1 core
PolyExpand – Farneback (2003)	52.00 %	59.94 %	53.32 %	100.00 %	1 s / 1 core
Pyramid-LK – yves Bouguet (2000)	71.84 %	78.32 %	72.91 %	100.00 %	1.5 min / 1 core

Table 7: **KITTI 2015 Optical Flow Leaderboard.** Numbers correspond to percentages of bad pixels according to the 3px/5% criterion defined in Menze & Geiger (2015) in background (bg), foreground (fg) or all regions. The methods below the horizontal line are older entries, serving as reference.

small to train large CNNs, Dosovitskiy et al. (2015) created the Flying Chairs dataset by rendering 3D chair models on top of images from Flickr. This first attempt to end-to-end optical flow learning demonstrated that it was possible to learn optical flow but could not reach state-of-the art performance on KITTI (Table 7) or Sintel. However, compared to methods performing at almost real-time they were the best performing. In contrast to the contracting and expanding networks of Dosovitskiy et al. (2015), Ranjan & Black (2016) present SpyNet, an architecture inspired by the coarse-to-fine matching strategy leveraged in traditional optical flow estimation techniques. Each layer of the network represents a different scale and only estimates the residual flow with respect to the warped image. This formulation allowed them to achieve similar performance as FlowNet while being faster. Being 96 % smaller than FlowNet one major contribution was the memory efficiency which makes it attractive for embedded systems. Ilg et al. (2016) present FlowNet2, an improved version of FlowNet, by stacking the architectures and fusing the stacked network with a subnetwork specialized on small motions. Similar to SpyNet, they also input the warped image into the stacked networks. However, each stacked network estimates the flow between the original frames instead of the residual flow as in SpyNet. In contrast to FlowNet and SpyNet, they use the FlyingThings3D dataset (Mayer et al. (2016)) consisting of 22k renderings of static 3D scenes with moving 3D models from ShapeNet dataset (Savva et al. (2015)). FlowNet2 performs on par with state-of-the-art methods on Sintel and outperforms all others on KITTI 2015 (Table 7) while being one of the fastest. They provide different network variants

for the spectrum between 8fps and 140fps allowing the trade-off between accuracy and computational resources.

Discussion: Robust optical flow methods need to handle intensity changes not caused by the actual motion of interest but by illumination changes, reflections and transparency. In real world scenes, repetitive patterns and occlusions are frequent sources of errors. While illumination changes have been addressed with novel data terms (Black & Anandan (1993); Vogel et al. (2013)) the problems caused by reflection, transparency, ambiguities and occlusions remain largely unsolved. In Figure 26 we show the accumulated error of the 15 best performing methods on KITTI 2015 (Menze & Geiger (2015)). The highest error can be observed for regions moving outside the image domain. Untextured, reflective and transparent regions also result in large errors in many cases. A better understanding of the world is necessary to tackle these problems. Semantics (Bai et al. (2016); Sevilla-Lara et al. (2016)) and learned high-capacity models (Dosovitskiy et al. (2015); Ranjan & Black (2016); Ilg et al. (2016)) have already proven to improve optical flow estimation by resolving ambiguities in the data. In addition, scene flow methods which jointly reason about flow and depth have demonstrated encouraging performance.

8.2. 3D Motion Estimation – Scene Flow

Stereo matching does not reveal any motion information, and optical flow from a single camera is not well constrained and lacks the depth information lost by the projection. On the other hand, humans are able to effortlessly integrate depth and
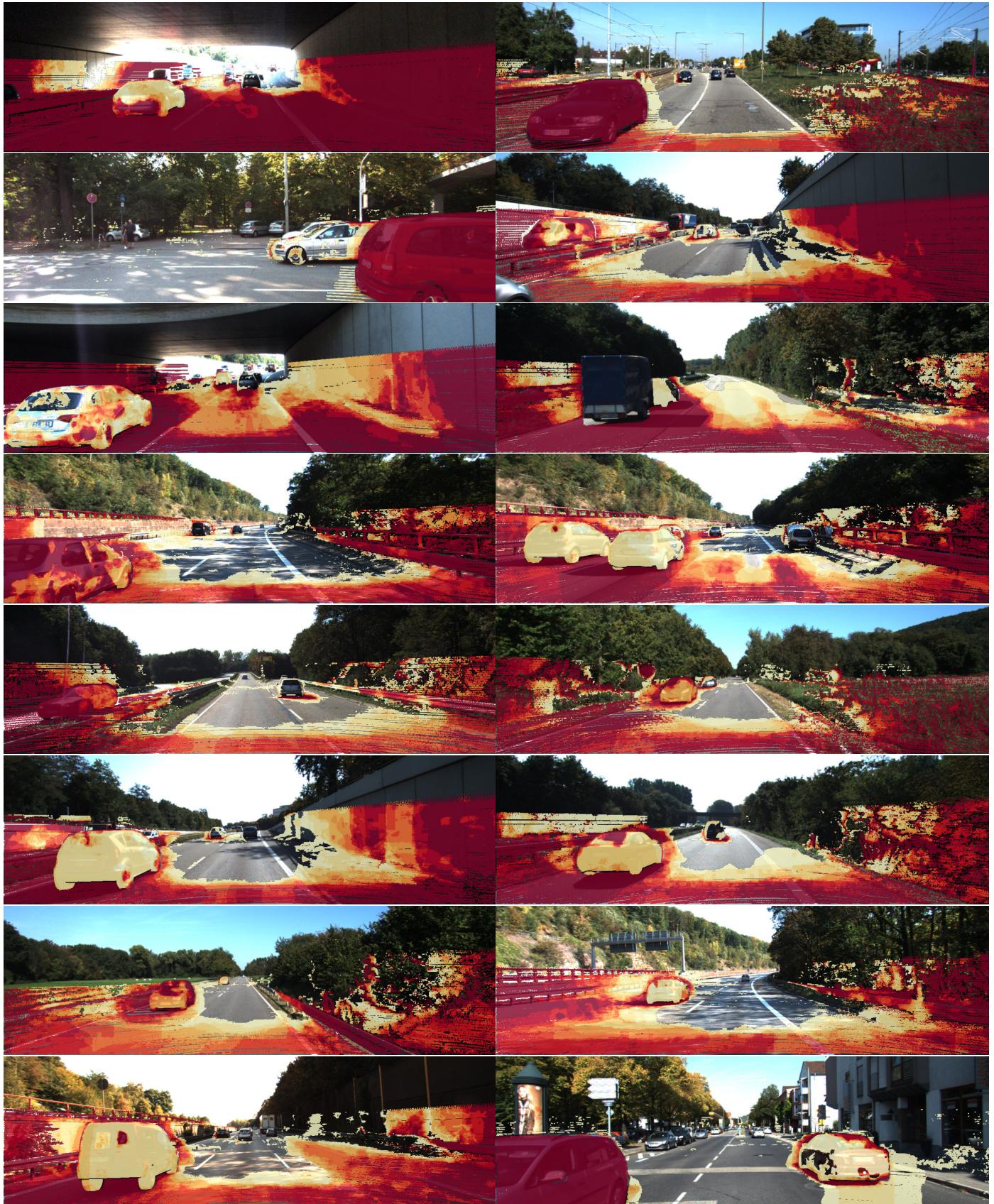
Figure 26: **KITTI 2015 Optical Flow Analysis.** Accumulated errors of 15 best-performing optical flow methods published on the KITTI 2015 Flow benchmark. Red colors correspond to regions where the majority of methods results in bad pixels according to the 3px/5% criterion defined in Menze & Geiger (2015). Yellow colors correspond to regions where some of the methods fail. Regions which are correctly estimated by all methods are shown transparent.
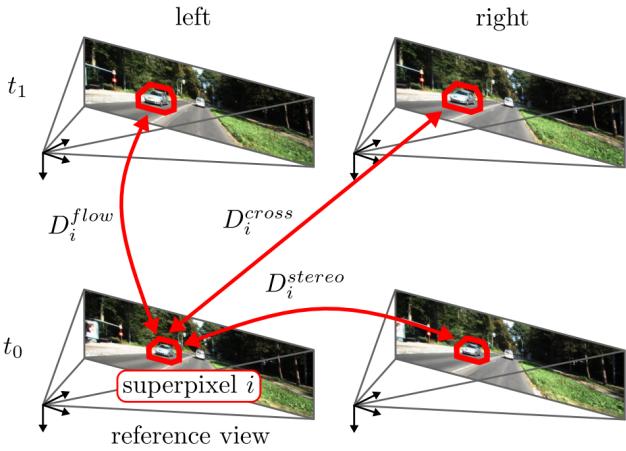
Figure 27: Scene flow. The minimal setup for image-based scene flow estimation is given by two consecutive stereo image pairs. Adapted from Menze & Geiger (2015).

Figure 28: Piecewise rigidity. The scene is modeled as a collection of rigidly moving planar segments. Adapted from Vogel et al. (2015).

motion cues from observations over time. That kind of reasoning is essential for many tasks in autonomous driving such as segmentation of moving objects in the 3D world. Scene flow generalizes optical flow to 3D, or alternatively, dense stereo to dynamic scenes. Given stereo image sequences, the goal is to estimate the three dimensional motion field that is a 3D motion vector for every point on every visible surface in the scene. The minimal setup for image-based scene flow estimation is given by two consecutive stereo image pairs as visualized in Figure 27. Establishing correspondences between the four images results in the 3D location of the surface point in both frames and hence fully describes the 3D motion of that surface point. A dense output is preferred, although there are some early sparse approaches for real-time purposes (Franke et al. (2005)). Scene flow shares some challenges with stereo and optical flow such as matching ambiguities in weakly textured regions and the aperture problem.

Variational Approaches: Following the seminal work by Vedula et al. (1999), the problem is traditionally formulated in a variational setting where optimization proceeds in a coarse-to-fine manner and local regularizers are leveraged to encourage smoothness in depth and motion. Wedel et al. (2008, 2011) propose a variational framework by decoupling the motion estimation from the disparity estimation while maintaining the stereo constraints. Starting from a precomputed disparity map at each time step, optical flow for the reference frame and disparity for the other view are estimated. The motivation for decoupling is mainly computational efficiency by choosing the optimal technique for each task. In addition, Wedel et al. (2011) propose a solution for varying lighting conditions based on residual images and provide an uncertainty measure which is shown to be useful for object segmentation. Rabe et al. (2010) integrate a Kalman filter to the decoupling approach for temporal smoothness and robustness.

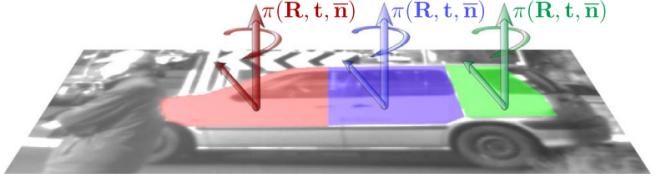**Piecewise Rigidity:** Similar to stereo and optical flow, prior assumptions about the geometry and motion can be exploited to better handle the challenges of the scene flow problem. Vogel et al. (2015) and Lv et al. (2016) represent the dynamic scene as a collection of rigidly moving planar regions as shown in Figure 28. Vogel et al. (2015) jointly recover this segmentation while inferring the shape and motion parameters of each region. They use a discrete optimization framework and incorporate occlusion reasoning as well as other scene priors in the form of spatial regularization of geometry, motion and segmentation. In addition, they reason over multiple frames by constraining the segmentation to remain stable over a temporal window. Their experiments show that their view-consistent multi-frame approach significantly improves accuracy in challenging scenarios, and achieves state-of-the-art performance on the KITTI benchmark (see Table 8). Using the same representation, Lv et al. (2016) focus on an efficient solution to the problem. They assume a fixed superpixel segmentation and perform optimization in the continuous domain for faster inference. Starting from an initialization based on Deep Matching, they independently refine the geometry and motion of the scene, and finally perform a global non-linear refinement using the Levenberg-Marquardt algorithm.

Piecewise Rigidity at the Object Level: Menze & Geiger (2015) also follow a slanted plane approach, but in addition to Vogel et al. (2015); Lv et al. (2016), they model the decomposition of the scene into a small number of independently moving objects and the background. By conditioning on a superpixelization, they jointly estimate this decomposition as well as the rigid motion of the objects and the plane parameters of each superpixel in a discrete-continuous CRF. Compared to Vogel et al. (2015); Lv et al. (2016), they leverage a more compact representation, implicitly regularizing over larger distances. They also present a new scene flow dataset by annotating dynamic scenes from the KITTI raw data collection using detailed 3D CAD models. They further present an extension of this model in Menze et al. (2015b) where the pose and 3D shape of the objects are inferred in addition to the rigid motion and the segmentation. In particular, they incorporate a deformable 3D active shape model of vehicles into the scene flow approach.

State-of-the-art: In Table 8, we show the ranking of methods on the KITTI scene flow 2015 benchmark (Menze & Geiger (2015)). The methods are compared according to the percentage of erroneous pixels. In particular, the columns show the percentage of stereo disparity outliers in first frame (D1), the percentage of stereo disparity outliers in second frame (D2), the percentage of optical flow outliers (Fl), and the percentage

Method	D1	D2	Fl	SF	Runtime
PRSM – Vogel et al. (2015)	4.27 %	6.79 %	7.28 %	9.44 %	300 s / 1 core
OSF – Menze & Geiger (2015)	5.79 %	7.77 %	8.37 %	10.63 %	50 min / 1 core
CSF – Lv et al. (2016)	4.57 %	10.06 %	13.71 %	16.33 %	80 s / 1 core
SGM+SF – Hirschmüller (2008)	6.84 %	15.60 %	22.24 %	25.43 %	45 min / 16 core
PCOF-LDOF – Derome et al. (2016)	8.46 %	20.99 %	18.83 %	29.63 %	50 s / 1 core
PCOF + ACTF – Derome et al. (2016)	8.46 %	22.00 %	22.80 %	33.02 %	0.08 s / GPU
SGM+C+NL – Hirschmüller (2008)	6.84 %	28.25 %	36.10 %	40.68 %	4.5 min / 1 core
SGM+LDOF – Hirschmüller (2008)	6.84 %	28.56 %	39.91 %	44.12 %	86 s / 1 core
DWBSF – Richardt et al. (2016)	20.12 %	34.46 %	39.87 %	46.02 %	7 min / 4 cores
GCSF – Cech et al. (2011)	14.21 %	33.41 %	47.00 %	53.95 %	2.4 s / 1 core
VSF – Huguet & Devernay (2007)	26.38 %	57.08 %	49.64 %	67.08 %	125 min / 1 core

Table 8: **KITTI 2015 Scene Flow Leaderboard.** Numbers correspond to percentages of bad pixels according to the 3px/5% criterion defined in Menze & Geiger (2015) for disparity in the first frame (D1), disparity in the second frame (D2), optical flow between both frames (Fl) as well as the combination of all criteria yielding the final scene flow metric (SF). The methods below the horizontal line are older entries, serving as reference.

of scene flow outliers (SF), i.e. outliers in either D0, D1 or Fl. The outlier ratio separately for foreground/background regions can be found on the website of the benchmark[30], it is omitted here for space reasons. The top performing methods (Vogel et al. (2015); Menze & Geiger (2015); Lv et al. (2016)) use the assumption of rigidly moving segments. In addition, Menze & Geiger (2015) model the motion of independently moving objects and perform better in Fl and SF on foreground regions, but it takes longer than the other two. Lv et al. (2016) achieve good results faster by focusing on efficient optimization in continuous domain. Derome et al. (2016) propose a two stage approach on GPU which runs several order of magnitude faster than the other methods. First they compute the static flow using stereo and visual odometry and correct the dynamic flow with a real-time optical approach Plyer et al. (2014).

Discussion: Scene flow estimation shares most of the challenges with stereo and optical flow, while integrating more information leading to better results. Ideally, methods should exploit depth and motion cues together to reason about dynamic 3D scenes. We show the accumulated errors of top 5 methods on KITTI scene flow benchmark in Figure 29. Car surfaces are the most problematic regions due to matching problems and independent motion of cars. Pixels close to the image boundary are another typical source of error, especially on the road surfaces in front of the car where large scale changes occur. Although local planarity and rigidity assumptions alleviate the problem, they are often violated due to complex geometric objects like vegetation, pedestrians or bicycles. Wrong estimation of planes, for example superpixels extending to multiple surfaces cause additional problems, especially at the boundaries of objects. Semantic image understanding could help with these issues, especially at the object level by segmenting car instances. Another way to integrate more information is to consider long-term temporal interactions.

8.3. Ego-Motion Estimation

The estimation of the ego-motion, the position and orientation of the car, is another fundamental problem to realize autonomous driving. Traditionally, this problem was addressed in wheel odometry with wheel encoders, which measure the rotation of the wheel, by integrating the measurements over time. These methods suffer from wheel slip in uneven terrain or adverse conditions and can not recover from errors in the measurements. Visual odometry or LiDAR-based odometry techniques, which estimate ego-motion from images or laser range measurements, became popular because they are less affected by these conditions and can correct estimation errors by recognizing already visited places which is called loop closure (Section 8.4.1). A detailed tutorial to this topic was presented by Scaramuzza & Fraundorfer (2011) and Fraundorfer & Scaramuzza (2011).

Formulation: In visual odometry the goal is to recover the full trajectory of one camera or a camera system from images. This is incrementally done by estimating the relative transformation between the camera positions at two time steps and accumulating all transformations over time to recover the full trajectory. The incremental approach is illustrated in Figure 30. The different methods can be divided into two categories: feature-based methods, that extract an intermediate representation (features) from raw measurements, and direct formulations, that directly operates on raw measurements. Feature-based methods typically work only in environments conforming the used feature type. Especially in man-made environments, important information about straight and curved edges is discarded considering keypoints. In contrast, direct methods leverage the gradient information of the whole image. Therefore, these methods usually achieve higher accuracy and robustness in environments with little keypoints. The field was dominated by feature-based methods since they typically were more efficient but direct formulations have recently grown in popularity. In feature-based and direct formulations, the extracted representation or raw measurements are usually used as input in a probabilistic model to compute unknown hidden model parameters such as the camera motion or a world model. A Maximum Likelihood

---

[30]`http://www.cvlibs.net/datasets/kitti/eval_scene_flow.php`

Figure 29: **KITTI 2015 Scene Flow Analysis.** Accumulated errors of 15 best-performing scene flow methods published on the KITTI 2015 Scene Flow benchmark. Red colors correspond to regions where the majority of methods results in bad pixels according to the 3px/5% criterion defined in Menze & Geiger (2015). Yellow colors correspond to regions where some of the methods fail. Regions which are correctly estimated by all methods are shown transparent.
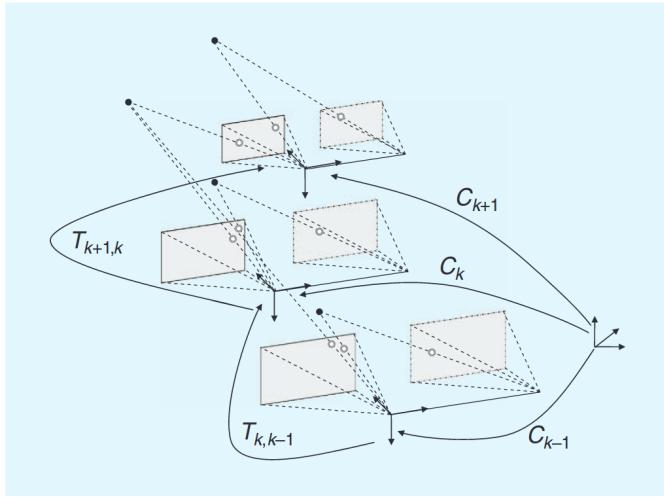
Figure 30: Illustration of the visual odometry problem by Scaramuzza & Fraundorfer (2011). The transformation $T_{k,k-1}$ between two adjacent camera positions (or positions of a camera system) is obtained using visual features. The accumulation of all transformations yields the absolute pose C_k with respect to the initial coordinate frame $k = 0$. Adapted from Scaramuzza & Fraundorfer (2011).

approach typically finds the model parameters that maximize the probability of obtaining the measurements.

Drift: The incremental approach greatly suffers from drift caused by the accumulation of estimation errors of the individual transformations. It is usually addressed with an iterative refinement over the last x images. This is done by reprojecting image points into 3D by triangulation and minimizing the sum of squared reprojection errors (sliding window bundle adjustment or windowed bundle adjustment). Another method to reduce drift is simultaneous localization and mapping (SLAM) (Lee et al. (2013a); Engel et al. (2015); Pire et al. (2015); Mur-Artal et al. (2015)) which jointly estimates the location and a map of the environment to recognize places that have been visited before. The detection of already mapped places is known as loop closure and is used to reduce the drift in the trajectory as well as the map and achieve global consistency. Some work focus on the loop closure detection in specific (Cummins & Newman (2008); Paul & Newman (2010); Lee et al. (2013b)) which will be discussed in detail in Section 8.4.1. These approaches are computationally expensive and a careful selection of the extracted features can already reduce the estimation error and drift. Kitt et al. (2010) for example use bucketing to obtain well distributed corner-like feature matches whereas Deigmoeller & Eggert (2016) use different heuristics on flow and depth estimation to reject non stable features.

2D-to-2D Matching: Depending on how corresponding points between two time steps are represented (2D or 3D), different methods must be used to obtain the camera transformation. In case of 2D feature matches (2D-to-2D) the essential matrix can be estimated which represents the epipolar geometry between the two cameras. The translation and rotation can directly be extracted from the essential matrix. The eight-point algorithm (Longuet-Higgins (1981)) is a simple solution working with calibrated and uncalibrated cameras whereas the five-point algorithm (Nistér (2004)) is a minimal case solution which only applies to the scenario of calibrated cameras. Scaramuzza et al. (2009) estimate the essential matrix from monocular images with only one 2D feature correspondence using non-holonomic constraints of wheeled vehicles imposing a restrictive motion model. Lee et al. (2013a) extend this idea to a novel two point minimal solution that is able to obtain the metric scale using a multi-camera system. In contrast to the non-holonomic constraints, Lee et al. (2014) assume the vertical directions to be known (from an Inertial Measurement Unit) and propose a minimal four-point and linear eight-point algorithm for a multi-camera system. Kitt et al. (2010) estimate the ego-motion using trifocal tensor which relates features between three images. Using these algorithms within RANSAC all 6 degrees of freedom can be robustly obtained in these special scenarios. The number of iterations necessary to guarantee that a correct solution is found with RANSAC depends on the number of points from which the model can be instantiated. Therefore, a reduced number of correspondences will reduce the number of iterations and the runtime of the approach.

3D-to-2D Matching: In the case of 3D features at the previous time step and 2D image features at the current time step (3D-to-2D) the transformation is estimated from stereo data (or triangulation when using monocular images). Geiger et al. (2011) present a real-time 3D reconstruction approach using visual odometry. They detect sparse features using blob, corner detector and estimate the ego-motion by minimizing the reprojection error. The estimation is refined with a Kalman filter while the dense 3D reconstruction is obtained by triangulating the image points. In contrast, Engel et al. (2013) continuously estimate a semi-dense inverse depth map to do real-time visual odometry with a monocular camera. The depth is estimated using multi-view stereo for pixel with non-eligible gradients and is represented by a Gaussian probability distribution. The depth estimation is propagated from frame to frame and the transformation is estimated using whole-image alignment. With this semi-dense formulation they achieve comparable performance to fully dense methods while not requiring a depth sensor. Engel et al. (2016) present a direct sparse approach for monocular visual odometry. They use a fully directed probabilistic model and jointly optimize all model parameters (camera poses, camera intrinsics, inverse depth).

3D-to-3D Matching: When dealing with 3D correspondences (3D-to-3D), the transformation can be obtained by aligning the two sets of 3D features. In case of visual odometry the extracted features from images are projected into 3D using depth whereas LiDAR-based approaches such as Zhang & Singh (2014, 2015) directly obtain the 3D points from the sensor. The triangulated 3D points from stereo will exhibit a large anisotropic uncertainty due to the small baseline and the quadratic increase of errors with respect to distance. Thus it is more natural to minimize reprojection errors in the images where error statistics can be approximated more easily while laser-based approaches do not suffer from this problem and thus can be optimized more easily in 3D space.

Only few datasets exist for visual odometry and most are too short or consists of low quality imagery. The KITTI benchmark Geiger et al. (2012b) discussed in Section 2 provides a large dataset of challenging sequences and evaluation metrics. We provide the KITTI leaderboard of monocular approaches in Table 9, stereo approaches in Table 10 and LiDAR-based approaches is provided in Table 11. The performance is measured with the average translational and rotational error for all possible subsequences of length $(100, \ldots, 800)$ meters.

Monocular Visual Odometry: Monocular visual odometry methods can recover the motion only up to a scale factor. The absolute scale can then be determined by computing the size of objects in the scene, from motion constraints, or integration with other sensors. The eight-point method proposed by Longuet-Higgins (1981) performs poorly in the presence of noise, in particular with uncalibrated cameras. Mirabdollah & Mertsching (2014) investigated the second order statistics of the essential matrix to reduce the estimation error with the eight-point method. They use the Taylor expansion up to the second order terms to obtain a covariance matrix that acts as regularization term along with the coplanarity equations. The drifting problem is particularly difficult in monocular visual odometry because of the lack of depth information. With a ground plane estimation in a real-time monocular SfM system Song & Chandraker (2014) deal with scale-drift and improve upon Mirabdollah & Mertsching (2014) results in Table 9. They combine multiple cues with learned models to adaptively weight per-frame observation covariances for ground plane estimation. Mirabdollah & Mertsching (2015) present a real-time and robust monocular visual odometry approach using the iterative five-point method. They obtain the location of landmarks with uncertainties using a probabilistic triangulation method and estimate the scale of the motion with low quality features on the ground plane. With this approach they outperform all monocular visual odometry methods in Table 9. Since the KITTI dataset require metric output the scale estimate has a strong impact on the performance of the approaches.

Stereo Visual Odometry: Stereo visual odometry methods do not have the problem of estimating the scale because it is directly known from the baseline between the cameras. In addition, they allow to deal with the drifting problem with a joint formulation of ego-motion estimation and mapping. Therefore, stereo methods are typically outperforming monocular methods on the KITTI dataset (see Table 9 and Table 10). Engel et al. (2015) propose a real-time large-scale direct SLAM algorithm that couples temporal multi-view stereo with static stereo from a camera setup (Figure 31). This allows them to estimate depth of pixels that are under-constrained in static stereo while avoiding scale-drift that occurs using multi-view stereo. The images are directly aligned based on photoconsistency of high contrast pixel. Pire et al. (2015) divide the problem into camera tracking and map optimization that can be run in parallel. While sharing the same map the tracking task matches features, creates new points and estimates the camera pose whereas the map op-
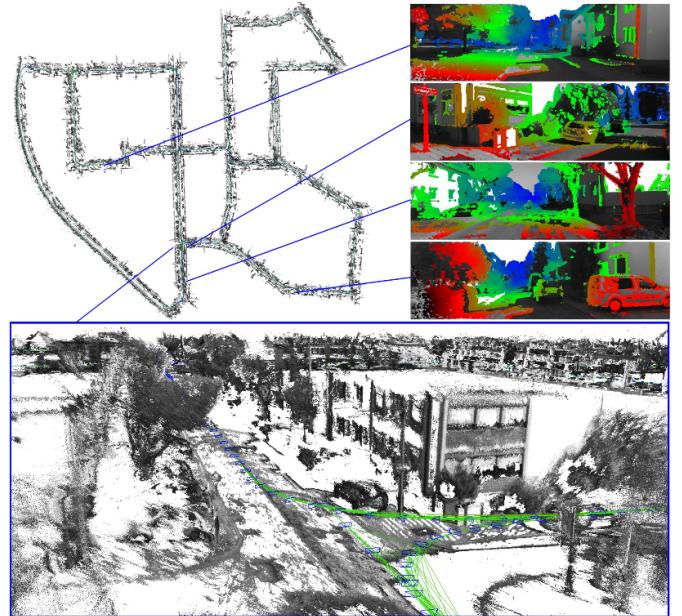

Figure 31: Stereo LSD-SLAM by Engel et al. (2015) computes accurate camera movement as well as semi-dense probabilistic depth maps in real-time. The depth visualization uses blue for far away scene points and red for close objects. Adapted from Engel et al. (2015).

timization refines the map with bundle adjustment. This formulation allows them to achieve the same performance while being faster. Deigmoeller & Eggert (2016) follow a very different way by relying exclusively on pure measurements as mentioned before. With the estimation of scene flow on Harris corners and rejection of features with different heuristics they outperform the two SLAM approaches in terms of translational error but have the highest rotational error and runtime in Table 10.

Persson et al. (2015) propose a stereo visual odometry system for automotive applications based on techniques from monocular visual odometry. In particular, they use motion model predicted tracking by matching, similar to Song et al. (2013), and delayed outlier identification. They argue that stereo techniques outperform monocular techniques because the problem formulation is easier. Monocular techniques should be more refined and robust because they need to deal with intrinsically more difficult problem. This allows them to outperform the others in the translational error in Table 10. The two best performing methods decouple the estimation of the rotation and translation as there is a fundamental difference between their estimation. The translation is dependent on the depth in contrast to the rotation. Buczko & Willert (2016a) claim that errors from depth estimation affect the rotation estimation in a coupled formulation and can be avoided by decoupling them. Thus, they use an initial rotation estimation to decouple the rotational and translational optical flow. The resulting characteristics are then used to exclude outliers. Cvisic & Petrovic (2015) compute the motion with a separate estimation of the rotation using the five point and the translation using the three point method. They also present a modified IMU-aided version of the algorithm suitable for embedded systems.

Method	Translation	Rotation	Runtime
FTMVO – Mirabdollah & Mertsching (2015)	2.24 %	0.0049 [deg/m]	0.11 s / 1 core
MLM-SFM – Song & Chandraker (2014)	2.54 %	0.0057 [deg/m]	0.03 s / 5 cores
RMCPE+GP – Mirabdollah & Mertsching (2014)	2.55 %	0.0086 [deg/m]	0.39 s / 1 core
VISO2-M – Geiger et al. (2011)	11.94 %	0.0234 [deg/m]	0.1 s / 1 core
OABA – Frost et al. (2016)	20.95 %	0.0135 [deg/m]	0.5 s / 1 core

Table 9: **KITTI Monocular Odometry Leaderboard.** The numbers show relative translational errors and relative rotational errors, averaged over all subsequences of length 100 meters to 800 meters.

Method	Translation	Rotation	Runtime
SOFT – Cvisic & Petrovic (2015)	0.88 %	0.0022 [deg/m]	0.1 s / 2 cores
RotRocc – Buczko & Willert (2016a)	0.88 %	0.0025 [deg/m]	0.3 s / 2 cores
ROCC – Buczko & Willert (2016b)	0.98 %	0.0028 [deg/m]	0.3 s / 2 cores
cv4xv1-sc – Persson et al. (2015)	1.09 %	0.0029 [deg/m]	0.145 s / GPU
NOTF – Deigmoeller & Eggert (2016)	1.17 %	0.0035 [deg/m]	0.45 s / 1 core
S-PTAM – Pire et al. (2015)	1.19 %	0.0025 [deg/m]	0.03 s / 4 cores
S-LSD-SLAM – Engel et al. (2015)	1.20 %	0.0033 [deg/m]	0.07 s / 1 core
VoBa – Tardif et al. (2010)	1.22 %	0.0029 [deg/m]	0.1 s / 1 core
MFI – Badino et al. (2013)	1.30 %	0.0030 [deg/m]	0.1 s / 1 core
2FO-CC – Krešo & Šegvić (2015)	1.37 %	0.0035 [deg/m]	0.1 s / 1 core
VISO2-S – Geiger et al. (2011)	2.44 %	0.0114 [deg/m]	0.05 s / 1 core
VOFS – Kaess et al. (2009)	3.94 %	0.0099 [deg/m]	0.51 s / 1 core

Table 10: **KITTI Odometry Stereo Leaderboard.** The numbers show relative translational errors and relative rotational errors, averaged over all subsequences of length 100 meters to 800 meters. Methods below the horizontal line are older entries for reference.
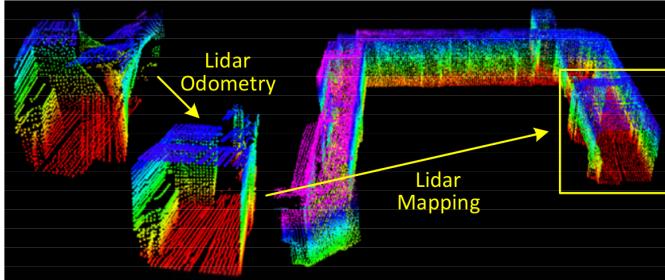

Figure 32: LOAM by Zhang & Singh (2014) matches two consecutive LiDAR scans (LiDAR Odometry) and registers the new scan to a map (LiDAR Mapping). Adapted from Zhang & Singh (2014).

Krešo & Šegvić (2015) observed that the camera calibration is critical for visual odometry and that the remaining calibration errors in pre-calibrated systems like KITTI have adversarial effects on the estimation results. They therefore propose to correct the calibration of the camera by exploiting the ground truth motion. The deformation field is recovered by optimizing the reprojection error of point feature correspondences in neighboring stereo frames under the groundtruth motion. Using the deformation field they obtained state-of-the-art results at the time.

LiDAR-based Odometry: The best performing methods on KITTI are using point clouds for ego-motion estimation (Table 11). Zhang & Singh (2014) split the SLAM problem into LiDAR-based odometry at high frequency with low fidelity and LiDAR-mapping at low frequency illustrated in Figure 32. The LiDAR-based odometry matches two consecutive LiDAR scans whereas the LiDAR-mapping matches and registers the new scan to a map. This results in low drift and computational complexity without the need for high accuracy range or inertial measurements. Zhang & Singh (2015) extend this work by combining visual odometry at high frequency with LiDAR-mapping at low frequency which allows them to further improve.

Discussion: Though several approaches have addressed the scale-drift problem in monocular visual odometry they cannot compete with approaches using 3D information on the KITTI dataset yet. While LiDAR provides the richest source of information for ego motion estimation, stereo-based methods show competitive results. In Figure 33 we visualize the average translational and rotational errors of the best performing visual odometry methods on the KITTI benchmark. The second row shows the translational error, the third row shows the rotational error while the last row shows the speed. The highest translational and rotational error can usually be obverse in strong turns. Furthermore, the error is correlated with speed and the amount of independently moving objects in the scene which lower the number of features in the background. While large errors can be observed for crowded highway scenes (second from right), only moderate errors occur when the highway is empty (right and second from left). Larger errors can also be observed in very narrow environments (fourth from right) where feature displacements are large. Overall, the most accurate motion estimation is achieved using 3D information, so far. However, stereo cameras are very cheap sensors in comparison to LiDAR laser scanners and stereo-based methods achieve competitive results.

Method	Translation	Rotation	Runtime
V-LOAM – Zhang & Singh (2015)	0.68 %	0.0016 [deg/m]	0.1 s / 2 cores
LOAM – Zhang & Singh (2014)	0.70 %	0.0017 [deg/m]	0.1 s / 2 cores
DEMO – Zhang et al. (2014)	1.14 %	0.0049 [deg/m]	0.1 s / 2 cores

Table 11: **KITTI Odometry LiDAR Leaderboard.** The numbers show relative translational errors and relative rotational errors, averaged over all subsequences of length 100 meters to 800 meters.

Figure 33: **KITTI Odometry.** From top-to-bottom: example image from the sequence, average translational error, average rotational error and speed. Averages are computed over 400 meter long trajectories and for the 15 best performing methods published on the KITTI website. Darker colors (i.e., red) indicate larger errors or higher speed. Each figure in a single row is normalized the same way to make them comparable.

8.4. Simultaneous Localization and Mapping (SLAM)

A detailed map of the environment is a commonly exploited prerequisite for path planning and navigation of an autonomous car. However, in places where a map is not provided or incomplete, the autonomous car needs to locate itself while generating the map. Further, the map needs to be updated continuously to reflect environmental changes over time. In this context, SLAM refers to the task of simultaneous estimation of the location of an agent while continuously building up a map of the environment. One particular challenge in autonomous driving, is that these systems need to handle large-scale environments in real-time.

Formulation: Traditionally, the map is represented by a set of landmarks, as for example image features. Early approaches to SLAM have addressed the problem using Bayesian formulations using extended Kalman filters (Smith et al. (1987)) or particle filters (Montemerlo et al. (2002)). Given the last state and current observations, the current state, represented by pose, velocity and the locations of the landmarks is recursively updated. However, this formulation is not applicable to large environments since the belief state and time complexity of the filter update grow quadratically in the number of landmarks in the map. The belief state represents all correlations between all pairs of variables which is $O(n^2)$ and whenever a landmark is observed the correlation to all other variables need to be updated with the same complexity. One solution for reducing complexity is a filtering technique based on a Graphical Model that maintains a tractable approximation of the belief state using a thin junction tree as proposed by Paskin (2003). However, it is known that filtering always produces an inconsistent map when applied on nonlinear SLAM problems (Julier & Uhlmann (2001)), which is usually the case when dealing with real data. In contrast, full SLAM approaches, such as graph-based or least-squares formulations, can provide exact solutions considering all poses at once. Kaess et al. (2008) propose an incremental smoothing and mapping approach based on fast incremental matrix factorization. They extend their work Dellaert & Kaess (2006) on factorizing the matrix of the nonlinear least-squares problem to an incremental approach that only recalculates entries which change in the matrix. Kaess et al. (2012) have introduced the Bayes tree, a novel data structure, to allow a better understanding of the connection between graphical model inference and sparse matrix factorization in SLAM. Factored probability densities are encoded in the Bayes tree which naturally maps to a sparse matrix.

Environmental Changes: A major challenge in SLAM are changes in the environment that might not be represented by a map. To alleviate this problem, Levinson et al. (2007) create a map only consisting of features that are very likely to be static. Using 3D LiDAR they retain only flat surfaces and ob-

Figure 34: Loop closure with appearance-based matching overlaid on an aerial image by Cummins & Newman (2008). Two images that are matched with a probability bigger than 99% are marked as red. Adapted from Cummins & Newman (2008).

Figure 35: Example models of Frahm et al. (2010) from Rome (left) and Berlin (right) computed in less than 24 hours. Adapted from Frahm et al. (2010).

tain an infrared reflectivity map of overhead views of the road surface. The map is then used to locate a vehicle with a particle filter in real-time. Levinson & Thrun (2010) extend this work considering maps as probability distributions over environment properties instead of a fixed representation. Specifically, every cell of the probabilistic map is represented as its own Gaussian distribution over remittance values. This allows them to represent the world more accurately and localize with fewer errors. In addition, they can use offline SLAM to align multiple passes of the same environment at different time to build an increasingly robust understanding of the world.

8.4.1. Loop Closure Detection

The relocalization in already mapped areas is an important subproblem of SLAM, referred to as loop closure detection. Relocalization is used to correct drifts in the trajectory and inaccuracies in the map caused by drift. Cummins & Newman (2008) present a probabilistic approach for the recognition of places based on their appearance. They learn a generative model of place appearances using bag-of-words because distinctive combinations of visual words will often arise from common objects. The generative model is robust and works even in visually repetitive environments. The performance of the approach is demonstrated on a self recorded dataset and visualized in Figure 34. Paul & Newman (2010) extend this idea by incorporating distance between words coupled to the observation of pairs of visual words with a random graph. The random graph models the pairwise distance between words besides their distribution of occurrences. In contrast, Lee et al. (2013b) show that the relative pose with metric scale between two loop-closing pose-graph vertices can directly be obtained from the epipolar geometry of a multi-camera system with overlapping views. They simplify the problem using a planar constraint on the motion of a car and estimate the loop-constraint using least squares optimization.

LiDAR-based: Image-based loop closure detection can become unreliable in case of strong illumination changes or strong viewpoints changes. In contrast, LiDAR-based localization is not affected by changes in illumination and does not suffer as

much from changes in viewpoint due to the captured 3D geometry. Dubé et al. (2016) propose a loop closure detection algorithm based on matching 3D segments. Segments from the point cloud are extracted and described using a combination of descriptors. Matching of segments is performed by obtaining candidates with kd-tree search in feature space and estimating the matching score of the candidates with a random forest.

8.4.2. Visual SLAM

Lategahn et al. (2011) propose a dense stereo visual SLAM method that estimates a dense 3D map. Using a sparse visual SLAM system, they obtain the pose and a sparse map. For the dense 3D map, they compute a dense representation from stereo in a local coordinate system and continuously update the map by tracking the local coordinate systems with the sparse SLAM system. Engel et al. (2014) extend their semi-dense method for visual odometry (Engel et al. (2013)) by performing image alignment and loop closure detection using a formulation based on optimizing the similarity transformation. Semi-dense depth is estimated using multi-view stereo from small baselines to create and refine a semi-dense map using pose graph optimization. The fusion of visual and inertial cues proposed by Leutenegger et al. (2013) takes advantage of their complementary nature. Instead of filtering, they use a non-linear optimization approach and integrate the IMU error with the reprojection error of landmarks into a joint cost function. Mur-Artal et al. (2015) use the ORB features proposed by Rublee et al. (2011) for tracking, mapping, relocalization and loop closure. They combine methods from loop detection (Gálvez-López & Tardós (2012)), loop closing (Strasdat et al. (2010, 2011)) and pose graph optimization (Kümmerle et al. (2011)) into one system.

8.4.3. Mapping

For autonomous driving applications, metric and semantic maps at different level of details are required to solve different tasks. Metric maps allow accurate localization whereas semantic maps can provide problem specific information such as parking areas for automated parking. Those maps can also be generated offline with computationally expensive methods and later incorporated into an autonomous driving system.

Metric Maps: The Google Street View project (Anguelov et al. (2010)) is a prominent example for a large collection of panoramic imagery in cities around the world. For collecting

the dataset, they estimate the pose in a Kalman-filter-based approach fusing data from GPS, wheel encoder and inertial navigation. Estimation at 100 Hz allows to accurately match image pixels from 15 small cameras to 3D rays from a laserscanner. The pose estimates are refined with a probabilistic graphical model of the network that represents all known roads and intersections in the world. From the image and laserscan data, they reconstruct the scene and obtain photorealistic 3D models by robustly fitting coarse meshes. Frahm et al. (2010) propose a dense 3D reconstruction approach from Internet-scale photo collections. Geometric relationships between the images are estimated using a combination of 2D appearance, color and 3D multi-view geometry constraints. They obtain the dense geometry of the scene via fast plane sweeping stereo and a depth map fusion approach. Exploiting the appearance and geometry constraints, they present a highly parallel approach which allows to process 3 million images within a day on a single computer. Figure 35 shows two example models reconstructed from Flickr images of Rome and Berlin. For autonomous driving applications, it is often sufficient to map the road surface in 2D (i.e., in bird's eye view) which allows for localization with respect to features on the road such as road markings or imperfections in the road surface. Geiger (2009) present an approach for road mosaicing in dynamic environments to create obstacle-free bird eye views. The road surface is extracted using optical flow on Harris corners and approximated by a plane. This allows to describe the mapping between the images with homographies. The road images are finally combined using multi-band blending.

Semantic Maps: All methods discussed so far focus on creating metric maps ignoring semantic information. However, for tasks like automated parking, a semantic map that is updated jointly with the metric map is necessary. Grimmett et al. (2015) fuse semantic and metric maps for vision-only automated parking. They update the map with static and dynamic labels and use active learning for lane, parking space and pedestrian crossings detection.

8.5. Localization

Localization is a well-studied problem in both robotics and vision, covering a broad range of techniques from indoor localization of a robot using noisy sensory measurements to locating where a picture was taken in the entire world. From an autonomous driving perspective, the main task is to precisely localize the ego-vehicle on a map. Localization is also an important subroutine of SLAM approaches, where it is used for detecting loop-closures and correcting drift when mapping the environment, see Section 8.4.1.

Localization can be performed using either sensors like a GPS system or visual information based on images. Using GPS alone typically provides an accuracy around 5 m. Although centimeter-level precision is possible in open spaces using combinations of sensors as in KITTI car (Geiger et al. (2012b)), it is often rendered infeasible in traffic scenes with several disturbing effects such as occlusions by vegetation and buildings

or multi-path effects due to reflections. Therefore, image-based localization independent of satellite systems is highly relevant.

Early image-based techniques (Li et al. (2009); Zheng et al. (2009)) approach the problem as classification into one of a predefined set of places which are referred as "landmarks". Others (e.g. Hays & Efros (2008)) create a database of images with known locations and formulate the localization as an image retrieval problem. These methods require a similarity measure to compare images based on local or global appearance cues. The larger the database, the more difficult the localization task becomes. Challenges include appearance changes, similar looking places, and the changes due to viewpoint or position.

Survey: Lowry et al. (2016) provide a comprehensive review of the current state of place recognition research. They first define what qualifies as a place in the context of robotic navigation by referencing to studies in psychology and neuroscience. Then, they review ways of describing a place using local or global descriptors and/or metric range information. They also provide a taxonomy based on the level of physical abstraction in the map and whether or not metric information is included in the place description. They further discuss how place recognition solutions can implicitly or explicitly account for appearance change within the environment and finally provide some future directions with respect to advances in deep learning, semantic scene understanding, and video description.

Monte Carlo Methods: The problem of map localization has been traditionally approached using Monte Carlo methods which recover the probability distribution over the agents pose by drawing a set of samples. Dellaert et al. (1999) define indoor localization in two steps, global position estimation and local position tracking over time. Instead of modeling the probability density function itself, they represent uncertainty by maintaining a set of samples and update the representation over time using Monte Carlo methods. This allows them to model arbitrary multimodal distributions in a memory efficient way. Outdoor localization is in general more challenging compared to the indoor localization task due to its scale and often unreliable sensor information such as GPS failures. Oh et al. (2004) use semantic information available in maps to compensate for the failure cases of GPS sensors. By exploiting knowledge about the environment, they assign probabilities to the target zones on the map, such as zero probability to the buildings. They incorporate these map-based priors in the particle filter formulation to bias the motion model toward areas of higher probability.

Metric, Topological, Topometric: Visual localization techniques are commonly classified into metric and topological methods. Metric localization is achieved by computing the 3D pose with respect to a map. Topological localization approaches provide a coarse estimate from a finite set of possible locations which are represented as nodes in a graph which are connected by edges that link them according to some distance or appearance criteria. Metric localization can be very accurate, but is usually not suitable for long sequences, while topological localization may be more reliable, but only provides rough estimates. Badino et al. (2012) propose a topometric approach as a com-

bination of topological and metric localization to provide geometrically accurate localization using graph-based methods. In contrast to topological methods, the graph is more fine-grained and each node corresponds to a metric location without a semantic meaning. During mapping phase, the graph is constructed using the vehicle position from GPS at fixed distance intervals and associating visual or 3D features to the corresponding graph node. At runtime, real-time localization is performed using a Bayes filter to estimate the probability distribution of the vehicle position along the route by matching features extracted from the sensor data to the map's feature database. Brubaker et al. (2016) also leverage a graph-based representation. In contrast to traditional localization approaches, however, they do not require a visual feature database of the environment, but instead directly build this graph from road networks extracted from OpenStreetMap. They further propose a probabilistic model which allows to infer a distribution over the vehicle location using visual odometry measurements. For tractability in very large environments, they leverage several analytic approximations for efficient inference yielding higher stability compared to particle-based filtering techniques which suffer from particle depletion when ambiguities persist over long periods.

Scale and Accuracy: For the problem of localization, the scale of the target area is a distinctive property to compare different approaches and is related to the accuracy achieved. Both scale and accuracy depend on the methodology used, such as map-based approaches (Brubaker et al. (2016)) which might suffer from the errors on the map and descriptor-based approaches (Badino et al. (2012); Schreiber et al. (2013)) using global or local descriptors. While the descriptor based method of Badino et al. (2012) achieves an average localization accuracy of 1 m over an 8 km route, the road network based localization approach of Brubaker et al. (2016) attains an accuracy of 4 m on a 18 km^2 map containing 2,150 km of drivable roads. Schreiber et al. (2013) point out that the required precision for autonomous driving and future driver assistance systems is in the range of a few centimeters and present a feature-based localization algorithm which can achieve this on approximately 50 km of rural roads. They approach the problem from the perspective of lane recognition. In a separate drive, they create a highly accurate map that contains road markings and curbs. While driving, they detect and match them to the map in order to determine the position of the vehicle relative to the markings.

Structure-based Localization: While the output of traditional localization approaches is either a rough camera position or a distribution over positions, a more recent line of work which is known as "structure-based localization" aims to estimate all camera matrix parameters, including position, orientation, and camera intrinsics. Localization is realized as a 2D-to-3D matching problem where the 2D points on the images are matched to a large, geo-registered 3D point cloud and the pose is estimated with respect to correspondences as shown in Figure 36.

In structure-based approaches, the pose estimate provides a powerful geometric constraint for validating the location estimate. However, a straightforward solution, for example di-
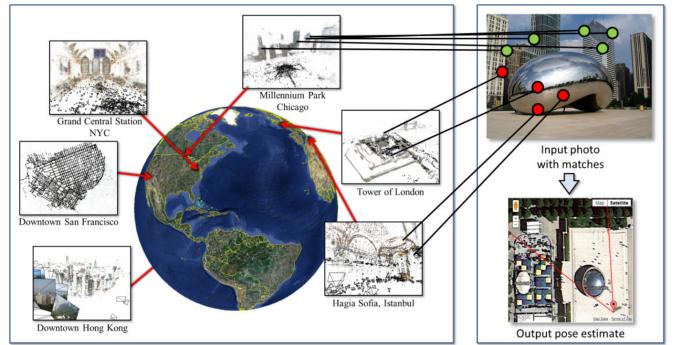

Figure 36: Localization. A query image is matched to a database of georeferenced structure from motion point clouds assembled from photos of places around the world (left). In structure-based approaches, the goal is to compute the georeferenced pose of new query images by matching to this worldwide point cloud (right). Adapted from Li et al. (2012).

rect matching by approximate nearest neighbor search using SIFT features, would result in many incorrect matches. With growing model size, the discriminative power of the descriptors decreases and matching becomes more ambiguous. Consequently, RANSAC techniques have difficulty finding the correct pose. To address this issue, Li et al. (2012) find statistical co-occurrences of 3D model points in images, and then use them as a sampling prior for RANSAC to exploit co-visibility relations. In addition, they employ a bidirectional matching scheme, forward from features in the image to points in the database and inverse from points to image features. They show that the bidirectional approach performs better than forward or inverse matching alone. Besides ambiguities, the amount of memory required for storing the large number of descriptors contained in the model is another problem related to large scale. Model compression by reducing the number of points produces fewer matches and increases the number of images which cannot be localized. Instead, more recent methods (Sattler et al. (2015, 2016)) use quantization into a fine vocabulary where each descriptor is represented by its word ID. Sattler et al. (2015) separate the difficult problem of finding a unique 2D-3D matching into two simpler ones. They first establish locally unique 2D-3D matches using a fine visual vocabulary and a visibility graph which encodes the visibility relation between 3D points and cameras. Then, they disambiguate these matches by using a simple voting scheme to enforce the co-visibility of the selected 3D points. Their experiments show that matching based on a visual vocabulary can achieve state-of-the-art. Sattler et al. (2016) propose a prioritized matching scheme based on quantization, focusing on efficiency. They significantly accelerate 2D-to-3D matching by considering more likely features first and terminating the correspondence search as soon as enough matches are found.

Structure-based Localization using Deep Learning: Kendall et al. (2015) and Walch et al. (2016) use a convolutional neural network to regress the camera pose from a single RGB image in an end-to-end manner. The motivation for using CNNs for this task is to eliminate the problems caused by large textureless areas, repetitive structures, motion blur, and illumination changes

which can be challenging for feature based methods. In contrast to classical localization approaches whose runtime depends on several factors such as the number of features found in a query image or the number of 3D points in the model, the runtime of CNN-based approaches only depends on the size of the network. Kendall et al. (2015) modify GoogLeNet (Szegedy et al. (2015)) by replacing softmax classifiers with affine regressors and inserting another fully connected layer before the final regressor which can be used as a localization feature vector for further analysis. The final architecture, dubbed PoseNet is initialized by using the weights of classification networks trained on giant datasets such as ImageNet (Deng et al. (2009)) and Places (Zhou et al. (2014)). Further, it is fine-tuned on a new pose dataset which was automatically created by using SfM to generate camera poses from a video of the scene. Walch et al. (2016) use a similar approach, but in addition they spatially correlate each element of the output of the CNN with Long Short-Term Memory (LTSM) units by exploiting their memorization capabilities. This way, the network is able to capture more contextual information and outperform PoseNet in different localization tasks including large-scale outdoor, small-scale indoor, and a newly proposed large-scale indoor localization benchmark. Although CNN-based approaches cannot match the precision of state-of the-art SIFT-based methods (Sattler et al. (2016)), their importance becomes more apparent in indoor environments with large textureless surfaces and repetitive scene elements where SIFT-based method cannot produce enough matches to obtain correct SfM reconstructions.

Cross-view Localization: It is difficult to keep ground imagery around the world up to date, while it is much easier to establish live maps from aerial images and satellites. This gives rise to a new approach, geo-localization which tries to register ground-level images to aerial imagery. The underlying idea is to learn a mapping between ground-level and aerial image viewpoints to localize a ground-level query in an aerial image reference database. Lin et al. (2013) match ground-level queries to other ground-level reference photos as in traditional geolocalization, but then use the overhead appearance and land cover attributes of those ground-level matches to build sliding-window classifiers in the aerial and land cover domain. In contrast to previous methods, they can often localize a query even if it has no corresponding ground-level images in the database by learning the co-occurrence of features in different views. Lin et al. (2015) collect a cross-view patch dataset using range data and camera parameters from Google street views to warp the dominant building surface plane to appear approximately like a 45% aerial view. Inspired by the success of face verification algorithms using deep learning, they train a Siamese network to match cross-view pairs of the same location. Workman et al. (2015) introduce another massive cross-view dataset. They first use CNNs for extracting ground-level image features and then, they learn to predict these features from aerial images of the same location. This way, the CNN is able to extract semantically meaningful features from aerial images without manually specifying semantic labels. They conclude that the cross-view localization approach can obtain a precise estimate of the
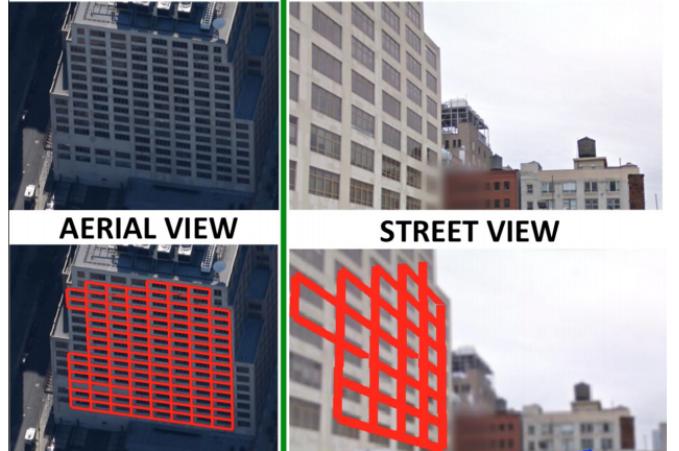

Figure 37: Aerial to street-view matching. The repeating patterns of buildings can yield valuable information for matching using regularity-driven approaches. Adapted from Wolff et al. (2016).

geographic locations which are distinctive from above. Otherwise, it can be used as a pre-processing step to a more expensive matching process.

Cross-view Localization: Buildings: There are methods specialized to building facades in cross-view matching. The repeating patterns can yield a valuable matching indicator for regularity-driven approaches (Figure 37). By combining satellite and oblique bird's eye-view, Bansal et al. (2011) first extract building outlines and facades and then match the ground image to oblique aerial images based on a statistical description of the facade pattern. Wolff et al. (2016) define a matching cost function to compare a street view motif to an aerial view motif based on similarity of color, texture and edge-based context features.

Cross-view Localization: Reconstructions: Another line of work addresses the problem of geo-referencing a reconstruction by automatic alignment with a satellite image, floor plan, map, or other overhead view. Kaminsky et al. (2009) compute the optimal alignment between SfM reconstructions and overhead images using an objective function that matches 3D points to image edges and imposes free space constraints based on the visibility of points in each camera. Matching ground and aerial images directly is a difficult endeavor due to the large differences in their camera viewpoints, occlusions, and imaging conditions. Instead of seeking invariant feature detections, Shan et al. (2014) propose a viewpoint-dependent matching technique by exploiting approximate alignment information and underlying 3D geometry.

Semantic Alignment from LiDAR: Several companies acquire LiDAR data from scanners mounted on cars driving through cities to acquire 3D models of real-world urban environments. However, the accuracy of the 3D point positions acquired by the 3D scanners depends on the scanner poses predicted by GPS, inertial sensors, and SfM, which often fail in urban environments. These misalignments cause problems for point cloud registration methods. Yu et al. (2015) propose to align semantic features that can be matched robustly at different scales. By fol-

lowing a coarse-to-fine approach, they first successively align roads, facades, and poles, which can be matched robustly. In the following, they match cars and other small objects, which require better initial alignments to find correct correspondences. The use of semantic features provides a globally consistent alignment of LiDAR scans and their evaluation shows improvement over the initial alignments.

9. Tracking

In tracking, the goal is to estimate the state of one or multiple objects over time given measurements of a sensor. Typically, the state of an object is represented by its location, velocity and acceleration at a certain time. Tracking of other traffic participants is a very important task in autonomous driving. Consider for instance the braking distance of a vehicle which increases quadratically with it's speed. In case of a possible collision with other traffic participants, the system needs to react early enough. The trajectory of other traffic participants allows to predict the future location and anticipate possible collisions. In case of pedestrians and bicyclists, it is particularly difficult to predict the future behavior because they can abruptly change the direction of their movements. However, tracking in combination with classification of traffic participants allows to adapt the speed of the vehicle to the situation. In addition, tracking of other cars can be used for automatic distance control and to anticipate possible driving maneuvers of other traffic participants (such as take overs) early on.

Challenges: Tracking systems must cope with a variety of challenges. Often, objects are partially or fully occluded by other objects or themselves. The resemblance of different objects is another challenge, in particular for objects of the same class. The interaction of objects in case of pedestrians further increases the amount of occlusions and makes it difficult to track each individual object. Difficult lighting conditions and reflections in mirrors or windows pose additional challenges.

Formulation: Several types of sensors have been exploited to address the tracking problem, e.g. monocular cameras, stereo cameras and laser scanners. Traditionally, tracking is formulated as a Bayesian inference problem. In that formulation, the goal is to estimate the posterior probability density function of a state given the current observation and the previous state(s). The posterior is usually updated in a recursive manner with a prediction step using a motion model and a correction step using an observation model. In each iteration, the data association problem is solved to assign new observations to the tracked objects. Extended Kalman and particle filtering algorithms (Giebel et al. (2004); Breitenstein et al. (2011); Choi et al. (2013)) are widely used models in this context. Unfortunately, the recursive approach makes it hard to recover from detection errors and to track through occlusions because of missing observations. Therefore, non-recursive approaches, which optimize a global Energy function with respect to all trajectories in a temporal window, have gained popularity. However, the large number of possible target trajectories per object and
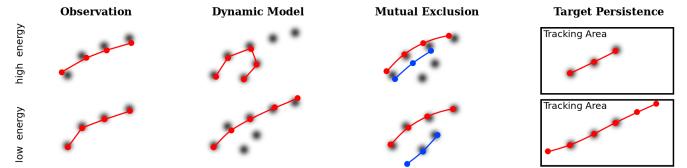

Figure 38: Components of the energy function proposed by Andriyenko & Schindler (2011). The upper and lower row show a configuration with a higher and smaller energy. The darker grey-values correspond to higher target likelihoods. Adapted from Andriyenko & Schindler (2011).

the large number of potential objects in a scene lead to a very large search space.

One way to approach this problem is to restrict the set of possible locations and solve the data association problem. Zhang et al. (2008) provide an elegant solution to this by casting the task as a min-cost flow problem which can be solved globally optimal in polynomial time in the presence of unary and pairwise potentials. They handle long-term inter-object occlusions by augmenting the network with an explicit occlusion model. Leibe et al. (2008b) focus on autonomous vehicle applications and propose a non-Markovian hypothesis selection framework for online tracking. This approach was extended through the integration of depth from stereo and odometry by Ess et al. (2009a).

Alternatively to discretization, continuous energy minimization approaches have been proposed. For this highly non-convex problem, Andriyenko & Schindler (2011) use a heuristic energy minimization scheme with repeated jump moves to escape week minima and better explore the variable-dimensional search space. The effect of different components of their energy function are illustrated in Figure 38. Milan et al. (2014) extend the continuous energy function of Andriyenko & Schindler (2011) to take into account physical constraints such as target dynamics, mutual exclusion, and track persistence. Assigning each observation to a certain target in data association is intrinsically in the discrete domain. Therefore, Andriyenko et al. (2012) argue that a joint discrete and continuous formulation describes the tracking problem more naturally. Their method alternates between solving the data association problem using discrete optimization with label costs and analytically fitting continuous trajectories while disregarding the label costs. Milan et al. (2013) propose a mixed discrete-continuous conditional random field model which specifically addresses mutual exclusion in the data association and the trajectory estimation. In data association each observation should be assigned to at most one target while in the trajectory estimation two trajectories should always remain spatially separated.

The most popular formulation for tracking multiple targets is tracking-by-detection. A classifier is used to detect objects of a certain object class which need to be associated with each other over time. This formulation has become very popular for multi object tracking since only relevant objects are tracked which allows to save computational resources. However, the tracking result is directly influenced by detection errors of the classifier.

Figure 39: The detections and corresponding top-down segmentations used by Leibe et al. (2008b) to learn an object-specific color model for tracking. Adapted from Leibe et al. (2008b).

Multiple Cue: For data association, the combination of different complementary cues has been observed to improve the robustness of tracking systems. Giebel et al. (2004) learn a spatio-temporal shape representation using distinct linear subspace models which can deal with appearance changes and combine shape, texture and depth from stereo in an observation model of a particle filter. Similarly, Gavrila & Munder (2007) integrate the same cues into a detection and tracking system with a cascade of modules. The stereo-based region of interest generation, shape-based detection, texture-based classification and stereo-based verification allow the system to focus on relevant image regions. They propose a novel mixture-of-experts architecture by weighting texture-based component classifiers by the outcome of shape matching. Choi et al. (2013) use a combination of detection systems, each specialized for a different task (pedestrian and upper body, face, skin color, depth-based shape and motion) in an appearance-based tracking approach. The response of all detection systems is combined in the observation likelihood to improve the matching between an observation and a track.

9.1. Tracking with Stereo

Some work has investigated a joint formulation for object tracking and stereo depth estimation to obtain the structure of the scene while estimating the trajectories of the objects in the scene. The structure of the scene allows the tracking system to focus on more plausible solutions. Leibe et al. (2007, 2008b) propose an approach integrating scene geometry estimation, 2D object detection, 3D localization, trajectory estimation and tracking. They learn object-specific color models using the detection and top-down segmentation of objects as illustrated in Figure 39. The structure of the scene is used to find physically plausible space-time trajectories and a final global optimization criterion takes object-object interactions into account to refine the 3D localization and trajectory estimation results. Ess et al. (2009a) jointly estimate the camera position, stereo depth, object detection and the pose of all objects over time using a graphical model. Thereby, the graphical model represents the interplay between the different components and incorporates object-object interactions.

Tracking-Before-Detection: In addition to facilitating the tracking problem, depth also allows to segment a scene into different objects independently of their class. In tracking-before-

detection these segmented class agnostic objects are directly considered as observations for the tracking formulation. This way the tracking system is independent of a classifier and thus also allows for tracking of objects which haven't been seen before or for which only little amounts of training data exists. Furthermore, motion information from the object's estimated trajectory can be used as another cue to detect a certain class of objects. Mitzel & Leibe (2012) extracts observations of objects by segmenting the scene using depth from stereo. With a compact 3D representation, they can robustly track known and unknown object categories. This representation also allows them to detect anomalous shapes such as carried items.

9.2. Pedestrian Tracking

Tracking and the detection of pedestrians is of particular importance for autonomous driving as mentioned before. Andriluka et al. (2008) combine advantages from detection and articulated human pose tracking in a single framework. They extend a state-of-the-art people detector with a limb-based structure model and model the dynamics of the detected limbs with a hierarchical Gaussian process latent variable model (hGPLVM). This allows them to detect people more reliably than approaches considering only one frame. In combination with a Hidden Markov Model (HMM) they can track people over very long sequences. They extend this idea in Andriluka et al. (2010) towards 3D pose estimation from monocular images. In the first stage they estimate 2D articulation and viewpoint of people and associate them across a small number of frames. This accumulated 2D image evidence is then used to estimate the 3D pose with a hGPLVM. The combination with a HMM allows to extend the tracks over longer time periods. This approach allows them to accurately estimate the 3D poses of multiple people from monocular images.

9.3. State-of-the-art

The most popular datasets for multi object tracking are PETS (Ferryman & Shahrokni (2009)), TUD (Andriluka et al. (2008)), ETHZ (Ess et al. (2008)), MOT (Leal-Taixé et al. (2015); Milan et al. (2016)) and KITTI (Geiger et al. (2012b, 2013)). Whereas PETS and TUD only provide data from a static observer, the others were acquired with a mobile platform which comes closer to the autonomous driving setting. In the MOTChallenge (Leal-Taixé et al. (2015)), the authors tackled the lack of a centralized benchmark for multi object tracking by presenting a new large dataset and evaluation methodology. The benchmark provides a detection ground truth for the tracking task which allows for comparing approaches based on their ability to track objects independent of errors caused by the detector. The leaderboard for methods using the detection ground truth is provided in Table 12 whereas the methods using a private detector are shown in Table 13. For the autonomous driving application, KITTI (Geiger et al. (2012b)) provides two benchmarks, one for tracking of cars (KITTI car) in Table 14 and the other for tracking of pedestrians in Table 15. Methods marked with an asterisk use Regionlet detections (Wang et al. (2015)) for an independent comparison of the tracking performance. In contrast

Method	MOTA	MOTP	MT	ML	IDS	FRAG	Hz
NOMT – Choi (2015)	46.4 %	76.6 %	18.3%	41.4%	359	504	2.6
JMC – Tang et al. (2016a)	46.3 %	75.7 %	15.5%	39.7%	657	1,114	0.8
oICF – Kieritz et al. (2016)	43.2 %	74.3 %	11.3%	48.5%	381	1,404	0.4
MHT_DAM – Kim et al. (2015)	42.9 %	76.6 %	13.6%	46.9%	499	659	0.8
LINF1 – Fagot-Bouquet et al. (2016)	41.0 %	74.8 %	11.6%	51.3%	430	963	4.2
EAMTT_pub – Sanchez-Matilla et al. (2016)	38.8 %	75.1 %	7.9%	49.1%	965	1,657	11.8
OVBT – Ban et al. (2016)	38.4 %	75.4 %	7.5%	47.3%	1,321	2,140	0.3
LTTSC-CRF – Le et al. (2016)	37.6 %	75.9 %	9.6%	55.2%	481	1,012	0.6
TBD – Geiger et al. (2014)	33.7 %	76.5 %	7.2%	54.2%	2,418	2,252	1.3
CEM – Milan et al. (2014)	33.2 %	75.8 %	7.8%	54.4%	642	731	0.3
DP_NMS – Pirsiavash et al. (2011)	32.2 %	76.4 %	5.4%	62.1%	972	944	212.6
GMPHD_HDA – m. Song & Jeon (2016)	30.5 %	75.4 %	4.6%	59.7%	539	731	13.6
SMOT – Dicle et al. (2013)	29.7 %	75.2 %	5.3%	47.7%	3,108	4,483	0.2
JPDA_m – Rezatofighi et al. (2015)	26.2 %	76.3 %	4.1%	67.5%	365	638	22.2

Table 12: **MOT16 Multi Target Tracking Leaderboard using Ground Truth Detections.** The metrics are detailed in Milan et al. (2016).

Method	MOTA	MOTP	MT	ML	IDS	FRAG	Hz
KDNT – Yu et al. (2016)	68.2 %	79.4 %	41.0%	9.0%	933	1,093	0.7
POI – Yu et al. (2016)	66.1 %	79.5 %	34.0%	20.8%	805	3,093	9.9
MCMOT_HDM – Lee et al. (2016a)	62.4 %	78.3 %	31.5%	24.2%	1,394	1,318	34.9
NOMTwSDP16 – Choi (2015)	62.2 %	79.6 %	32.5%	31.1%	406	642	3.1
SORTwHPD16 – Bewley et al. (2016)	59.8 %	79.6 %	25.4%	22.7%	1,423	1,835	59.5
EAMTT – Sanchez-Matilla et al. (2016)	52.5 %	78.8 %	19.0%	34.9%	910	1,321	12.2

Table 13: **MOT16 Multi Target Tracking Leaderboard using Private Detector.** The metrics are detailed in Milan et al. (2016).

to the MOTChallenge, the two separate datasets allow to focus the analyses on one object class and to deeply investigate the problems related to this class. In the Tables 12,13,14,15 we consider the two popular tracking measures, Multiple Object Tracking Accuracy (MOTA) and Multiple Object Tracking Precision (MOTP) introduced by Stiefelhagen et al. (2007), the ratio of mostly tracked (MT) and mostly lost trajectories (ML), number of ID switches (IDS) and of track segmentations (FRAG). Mostly tracked or lost trajectories are ground truth trajectories that are covered by an hypothesis at least 80% or at most 20% respectively.

On MOT16: With a focus on overcoming problems due to appearance changes of tracked objects Fagot-Bouquet et al. (2016) use a sparse representation-based appearance model in an energy minimization formulation. Such an appearance model defines a linear subspace using a small number of templates grouped in a dictionary to model the target appearance. Tang et al. (2016b) propose a minimum cost subgraph multicut formulation solving the spatial and temporal associations of observations while incorporating local pairwise features. The pairwise features are based on local appearance matching which is robust to partial occlusion and camera motion. This allows them to use an efficient algorithm that can handle long videos with many detections and to outperform Fagot-Bouquet et al. (2016) on MOT16. The best performing method on MOT16 using the provided detection ground truth was proposed by Levinkov et al. (2016). They consider a combinatorial optimization problem whose solution defines a decomposition and node labeling of

a graph. They solve this problem with a local search algorithm that converges monotonously to a local minimum. The multicut formulation of Tang et al. (2016b) can be identified as a special case of this formulation.

On KITTI: For the task of car tracking, Lenz et al. (2015) propose a computational and memory bounded version of the min-cost flow tracking formulation presented in Zhang et al. (2008). This approach achieves good accuracy and precision while being amongst the fastest approaches on KITTI car (Table 14).

Another online tracking approach was presented by Yoon et al. (2015) for tracking cars and pedestrians. In this work they address the problem of complex camera motion in which case conventional motion models do not hold. They are able to factor out the camera motion by constructing a relative motion network to describe the relative motion between objects. Exploiting a Bayesian formulation, they show the advantage of using multiple relative motion models and improve in comparison to Lenz et al. (2015). In KITTI pedestrian benchmark (Table 15) they are part of the best performing methods. A similar performance achieves the near-online multi-target tracking algorithm presented by Choi (2015) formulated as a global data association problem. Their main contribution is an Aggregated Local Flow Descriptor (ALFD) which encodes relative motion patterns. They can robustly match far distance detections regardless of the application. Using multiple feature cues, their method outperforms all the online tracking approaches on KITTI car.

Method	MOTA	MOTP	MT	ML	IDS	FRAG	Runtime
MCMOT-CPD – Lee et al. (2016a)	72.11 %	82.13 %	52.13 %	11.43 %	233	547	0.01 s / 1 core
MDP – Xiang et al. (2015a)	69.35 %	82.10 %	51.37 %	13.11 %	135	401	0.9 s / 8 cores
NOMT* – Choi (2015)	69.73 %	79.46 %	56.25 %	12.96 %	36	225	0.09 s / 16 cores
SCEA* – Yoon et al. (2016)	67.11 %	79.39 %	52.13 %	10.98 %	106	466	0.06 s / 1 core
LP-SSVM* – Wang et al. (2016)	66.35 %	77.80 %	55.95 %	8.23 %	63	558	0.02 s / 1 core
mbodSSP* – Lenz et al. (2015)	62.64 %	78.75 %	48.02 %	8.69 %	116	884	0.01 s / 1 core
DCO-X* – Milan et al. (2013)	55.49 %	78.85 %	36.74 %	14.02 %	323	984	0.9 s / 1 core
RMOT* – Yoon et al. (2015)	53.03 %	75.42 %	39.48 %	10.06 %	215	742	0.02 s / 1 core
NOMT – Choi (2015)	55.87 %	78.17 %	39.94 %	25.46 %	13	154	0.09 s / 16 core
ODAMOT – Gaidon & Vig (2015)	54.87 %	75.45 %	26.37 %	15.09 %	403	1298	1 s / 1 core
LP-SSVM – Wang et al. (2016)	51.80 %	76.93 %	35.06 %	21.49 %	16	430	0.05 s / 1 core
SCEA – Yoon et al. (2016)	51.30 %	78.84 %	26.22 %	26.22 %	17	468	0.05 s / 1 core
TBD – Geiger et al. (2014)	49.52 %	78.35 %	20.27 %	32.16 %	31	535	10 s / 1 core
CEM – Milan et al. (2014)	44.31 %	77.11 %	19.51 %	31.40 %	125	398	0.09 s / 1 core
DP-MCF – Pirsiavash et al. (2011)	35.72 %	78.41 %	16.92 %	35.67 %	2738	3239	0.01 s / 1 core
DCO – Andriyenko et al. (2012)	28.72 %	74.36 %	15.24 %	30.79 %	223	622	0.03 s / 1 core

Table 14: **KITTI Car Tracking Leaderboard.** The metrics are detailed in Geiger et al. (2012b).

Method	MOTA	MOTP	MT	ML	IDS	FRAG	Runtime
MCMOT-CPD – Lee et al. (2016a)	40.50 %	72.44 %	20.62 %	34.36 %	144	775	0.01 s / 1 core
MDP – Xiang et al. (2015a)	35.91 %	70.36 %	23.02 %	27.84 %	88	830	0.9 s / 8 cores
SCEA* – Yoon et al. (2016)	39.34 %	71.86 %	16.15 %	43.30 %	56	649	0.06 s / 1 core
NOMT* – Choi (2015)	38.98 %	71.45 %	26.12 %	34.02 %	63	672	0.09 s / 16 cores
RMOT* – Yoon et al. (2015)	36.42 %	71.02 %	19.59 %	41.24 %	156	760	0.02 s / 1 core
LP-SSVM* – Wang et al. (2016)	34.97 %	70.48 %	20.27 %	34.36 %	73	814	0.02 s / 1 core
NOMT-HM* – Choi (2015)	31.43 %	71.14 %	21.31 %	41.92 %	186	870	0.09 s / 8 cores
SCEA – Yoon et al. (2016)	26.02 %	68.45 %	9.62 %	47.08 %	16	724	0.05 s / 1 core
NOMT – Choi (2015)	25.55 %	67.75 %	17.53 %	42.61 %	34	800	0.09 s / 16 core
RMOT – Yoon et al. (2015)	25.47 %	68.06 %	13.06 %	47.42 %	81	692	0.01 s / 1 core
LP-SSVM – Wang et al. (2016)	23.37 %	67.38 %	12.03 %	45.02 %	72	825	0.05 s / 1 core
CEM – Milan et al. (2014)	18.18 %	68.48 %	8.93 %	51.89 %	96	610	0.09 s / 1 core
NOMT-HM – Choi (2015)	17.26 %	67.99 %	14.09 %	50.52 %	73	743	0.09 s / 8 cores

Table 15: **KITTI Pedestrian Tracking Leaderboard.** The metrics are detailed in Geiger et al. (2012b).

In contrast to the Bayesian and min-cost formulation, Xiang et al. (2015a) consider the tracking problem as a Markov decision process (MDP). They learn a policy for the MDP using reinforcement learning which corresponds to learning a similarity function for data association. With this approach, Xiang et al. (2015a) is one of the best performing methods on KITTI car. Lee et al. (2016b) combine a convolutional neural network based object and motion detector in a Bayesian filtering framework. They detect drift and occlusions using a changing point detection algorithm. In both KITTI benchmarks (Tables 14,15) this approach outperforms all others in accuracy (MOTA) and precision (MOTP).

9.4. Discussion

Reliable tracking-by-detection can only be achieved using reasonable object detections. The impact of the detection system can be observed when comparing the methods marked with and without asterisks in KITTI (Tables 14,15) or the MOT16 leaderboard of methods using ground truth detections in Table 12 and object detectors in Table 13. However, object de-

tectors are already discussed in Section 5.6, thus we focus this discussion on the tracking problem. Similarly to the detection problem, tracking pedestrians is more challenging than cars. The reason is that the motion of pedestrians is very hard to predict since they can abruptly change the direction whereas the motion of cars can be modeled easily. In real scenes, partial and full occlusion of cars or pedestrians can be observed frequently which causes detection failures. In these cases the tracking system needs to re-recognize the tracked objects which can be difficult because of changes in lighting conditions or similarity to other objects in the proximity. These problem cause reinitialization of trajectories which can be observed in the high number of fragments (FRAG) and ID switches (IDS) in MOT16 and KITTI. Furthermore, we note that so far most tracking systems are complex and no end-to-end multiple target tracking algorithm has been proposed in the literature. Bridging this gap from detection to tracking might be a promising avenue for future research.
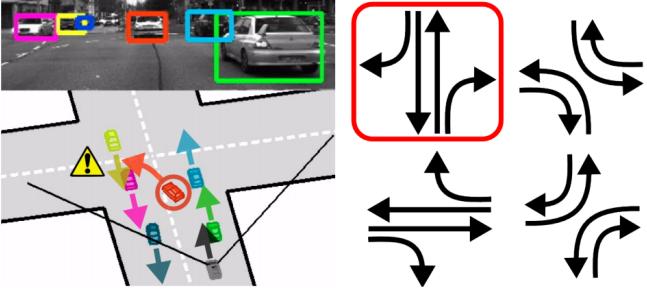
Figure 40: Scene understanding using traffic patterns. In Geiger et al. (2014), high-order dependencies between objects are ignored, leading to physically implausible inference results with colliding vehicles (left). Zhang et al. (2013) propose to explicitly account for traffic patterns to improve scene layout and activity estimation results (right, correct situation marked in red). Adapted from Zhang et al. (2013).
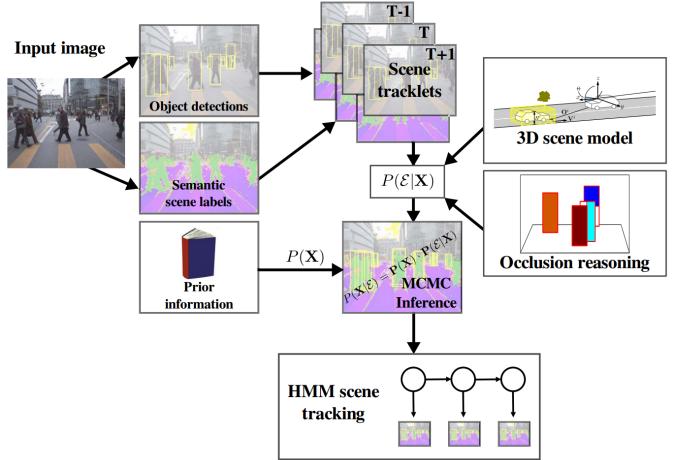

Figure 41: Overview of combined object detection and tracking system with explicit occlusion reasoning by Wojek et al. (2013). Adapted from Wojek et al. (2013).

10. Scene Understanding

One of the basic requirements of autonomous driving is to fully understand its surrounding area such as a complex traffic scene. The complex task of outdoor scene understanding involves several sub-tasks such as depth estimation, scene categorization, object detection and tracking, event categorization, and more. Each of these tasks describe particular aspect of a scene. It is beneficial to model some of these aspects jointly to exploit the relations between different elements of the scene and obtain a holistic understanding. The goal of most scene understanding models is to obtain a rich but compact representation of the scene including all its elements e.g., layout elements, traffic participants and the relations with respect to each other. Compared to reasoning in the 2D image domain, 3D reasoning plays a significant role in solving geometric scene understanding problems and results in a more informative representation of the scene in the form of 3D object models, layout elements and occlusion relationships. One specific challenge in scene understanding is the interpretation of urban and sub-urban traffic scenarios. Compared to highways and rural roads, urban scenarios comprise many independently moving traffic participants, more variability in the geometric layout of roads and crossroads, and an increased level of difficulty due to ambiguous visual features and illumination changes.

From Single Image to Video: In their pioneering work, Hoiem et al. (2007) infer the overall 3D structure of a scene from a single image. The surface layout is represented as a set of coarse geometric classes with certain orientations such as support, vertical, and sky. These elements are inferred by learning an appearance-based model for each class through multiple segmentations. Ess et al. (2009b) propose a more fine-grained approach both in terms of classification and representation using superpixels for recognizing the road and object types in a traffic scene. Liu et al. (2014) also use superpixels for single image depth estimation by retrieving similar images from a pool of images with known depth and modeling the occlusion relationships between superpixels. Although these methods show promising results when applied to a single image, motion in video sequences is a rich source of information es-

pecially in highly dynamic scenes. Kuettel et al. (2010) model spatio-temporal dependencies of moving agents in complex dynamic scenes by learning co-occurring activities and temporal rules between them. However, their approach assumes a static observer and the scene must be observed for a significant period of time before a decision can be made, therefore it is not applicable to autonomous systems. Geiger et al. (2014) jointly reason about the 3D scene layout of intersections as well as the location and orientation of vehicles in the scene using a probabilistic model. In this approach, the assumption that tracklets are independent can lead to implausible configurations such as cars colliding with each other. Zhang et al. (2013) resolve this issue by including high-level semantics into the formulation in the form of traffic patterns as shown in Figure 40.

Combined Object Detection and Tracking: Scene labeling is often combined with object detection and tracking to enable information flow between different but related tasks. Wojek & Schiele (2008a) detect vehicles and track them with a temporal filter based on a linear motion model. They also estimate the camera motion and propagate it to the next frame in a dynamic conditional random field model for joint labeling of object and scene classes. Wojek et al. (2010) extend joint reasoning to 3D by including pedestrians in the formulation. They propose a probabilistic 3D scene model that encompasses multi-class object detection, object tracking, scene labeling, and 3D geometric relations. The joint scene tracklet model over multiple frames improves performance for 3D multi-object tracking tasks without using stereo, however, this kind of approach is not capable of handling partially occluded objects. In order to solve this problem, Wojek et al. (2011, 2013) integrate multiple object part detectors into the 3D scene model for explicit object-object occlusion reasoning (Figure 41).

Other Representations: Apart from the 3D primitive based representations used by the aforementioned methods, there are other ways of representing a street scene. Seff & Xiao (2016) define a list of road layout attributes such as number of lanes,

drivable directions, distance to intersection, etc. They first automatically collect a large-scale dataset for these attributes by leveraging existing street view image databases and online navigation maps (e.g., OpenStreetMap). Based on this dataset, they train a deep convolutional network to predict each attribute from a single street view image. The goal is to reduce the dependency on high definition maps by serving as a backup in failure cases. Inspired by the prevalence of geometric structures outdoors, de Oliveira et al. (2016) represent the 3D structure with a set of planar polygons described by a support plane and a bounding polygon. Given 3D point cloud from LiDAR, they find the support plane by using RANSAC followed by a clustering of inliers to separate instances. For an incremental 3D representation of scenes over time, they evolve the representation as new data arrives. This is performed by a perpendicular and a longitudinal expansion of primitives to accommodate new point cloud data. Although a direct 3D representation of the scene from 3D measurements is usually not preferred due to the computational restrictions, their compact representation enables fast computation and updates while still being accurate.

11. End-to-End Learning of Sensorimotor Control

Current state-of-the-art approaches to autonomous driving are comprised of numerous models, e.g., detection (of traffic signs, lights, cars, pedestrians), segmentation (of lanes, facades), motion estimation, tracking of traffic participants, reconstruction. The results from these components are then combined in a rule based control system. However, this requires robust solutions to many open challenges in scene understanding in order to solve the problem of manipulating car direction and speed. As an alternative, several methods for end-to-end autonomous driving have been proposed in the literature recently. End-to-end autonomous driving is defined as driving using a self-contained system that maps from a sensory input, such as front-facing camera images, directly to driving actions such as steering angle.

Bojarski et al. (2016) propose an end-to-end deep convolutional neural network for lane following that maps images from the front facing camera of a car to steering angles, given expert data. Instead of directly learning the mapping from pixels to actions, Chen et al. (2015a) present an approach which first estimates a small number of human interpretable, pre-defined affordance measures such as the distance to surrounding cars. These predicted measures are then manually associated with car actions to enable a controller for autonomous driving.

Existing end-to-end learning methods maps pixels to actuation and directly mimics the demonstrated performance. However, the success of these methods is restricted to data collected in certain situations, in corresponding simulations or with specific calibrated actuation setup since the availability of public datasets for training is limited. Therefore, Xu et al. (2016) propose an alternative approach of exploiting large scale online datasets from uncalibrated sources to learn a driving model. Specifically, they formulate autonomous driving as a future egomotion prediction problem using a novel deep learning architecture that learns to predict the motion path given the present

agent state. Sharifzadeh et al. (2016) present a different paradigm based on Markov Decision Processes, for more promising performance when facing scenarios that are very different from the ones in the training data. Specifically, they apply Inverse Reinforcement Learning to extract the unknown reward function of the driving behavior. This allows them to handle new scenarios better compared to learning the state-action value pairs in a supervised way.

12. Conclusion

In this paper we provided a general survey on problems, datasets and methods in computer vision for autonomous vehicles. Towards this goal, we considered the historically most relevant literature as well as the state-of-the-art on several specific topics, including recognition, reconstruction, motion estimation, tracking, scene understanding and end-to-end learning. We discussed open problems and current research challenges in these topics using a novel in-depth qualitative analysis of the KITTI benchmark and considering other datasets. Our interactive online tool[31] allows an easy navigation through the surveyed literature with a visualization of our taxonomy using a graph. In future, we plan to keep the tool updated with relevant literature to provide an up-to-date overview of the field. We hope that our survey and tool will encourage new research and ease the entry in the field for beginners by providing an exhaustive overview.

References

References

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., & Süsstrunk, S. (2012). SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *34*, 2274–2282.

Álvarez, J. M., Gevers, T., LeCun, Y., & López, A. M. (2012). Road scene segmentation from a single image. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Alvarez, J. M., Gevers, T., & Lopez, A. M. (2010). 3d scene priors for road detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Álvarez, J. M., & López, A. M. (2011). Road detection based on illuminant invariance. *IEEE Trans. on Intelligent Transportation Systems (TITS)*, *12*, 184–193.

Anandan, P. (1989). A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision (IJCV)*, *2*, 283–310.

Andriluka, M., Roth, S., & Schiele, B. (2008). People-tracking-by-detection and people-detection-by-tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Andriluka, M., Roth, S., & Schiele, B. (2010). Monocular 3d pose estimation and tracking by detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Andriyenko, A., & Schindler, K. (2011). Multi-target tracking by continuous energy minimization. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Andriyenko, A., Schindler, K., & Roth, S. (2012). Discrete-continuous optimization for multi-target tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

---

[31]http://www.cvlibs.net/projects/autonomous_vision_survey

Anguelov, D., Dulong, C., Filip, D., Frueh, C., Lafon, S., Lyon, R., Ogale, A. S., Vincent, L., & Weaver, J. (2010). Google street view: Capturing the world at street level. *IEEE Computer*, *43*, 32–38.

Arbeláez, P. A., Pont-Tuset, J., Barron, J. T., Marqués, F., & Malik, J. (2014). Multiscale combinatorial grouping. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Arnab, A., & Torr, P. H. S. (2017). Pixelwise instance segmentation with a dynamically instantiated network. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Audebert, N., Saux, B. L., & Lefèvre, S. (2016). Semantic segmentation of earth observation data using multimodal and multi-scale deep networks. *arXiv.org*, *1609.06846*.

Badino, H., Franke, U., & Mester, R. (2007). Free space computation using stochastic occupancy grids and dynamic programming. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV) Workshops*.

Badino, H., Franke, U., & Pfeiffer, D. (2009). The stixel world - a compact medium level representation of the 3d-world. In *Proc. of the DAGM Symposium on Pattern Recognition (DAGM)*.

Badino, H., Huber, D., & Kanade, T. (2012). Real-time topometric localization. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Badino, H., Yamamoto, A., & Kanade, T. (2013). Visual odometry by multi-frame feature integration. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV) Workshops*.

Badrinarayanan, V., Budvytis, I., & Cipolla, R. (2014). Mixture of trees probabilistic graphical model for video segmentation. *International Journal of Computer Vision (IJCV)*, *110*, 14–29.

Badrinarayanan, V., Galasso, F., & Cipolla, R. (2010). Label propagation in video sequences. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Badrinarayanan, V., Kendall, A., & Cipolla, R. (2015). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv.org*, *1511.00561*.

Bai, M., Luo, W., Kundu, K., & Urtasun, R. (2016). Exploiting semantic information and deep matching for optical flow. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Bai, M., & Urtasun, R. (2016). Deep watershed transform for instance segmentation. *arXiv.org*, *1611.08303*.

Baker, S., Scharstein, D., Lewis, J., Roth, S., Black, M., & Szeliski, R. (2011). A database and evaluation methodology for optical flow. *International Journal of Computer Vision (IJCV)*, *92*, 1–31.

Ban, Y., Ba, S., Alameda-Pineda, X., & Horaud, R. (2016). Tracking multiple persons based on a variational bayesian model. In *Proc. of the European Conf. on Computer Vision (ECCV) Workshops*.

Bansal, M., Sawhney, H. S., Cheng, H., & Daniilidis, K. (2011). Geo-localization of street views with aerial image databases. In *Proc. of the International Conf. on Multimedia (ICM)*.

Bao, S., Chandraker, M., Lin, Y., & Savarese, S. (2013). Dense object reconstruction with semantic priors. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Behley, J., Steinhage, V., & Cremers, A. B. (2013). Laser-based segment classification using a mixture of bag-of-words. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*.

Benenson, R., Mathias, M., Timofte, R., & Gool, L. J. V. (2012). Pedestrian detection at 100 frames per second. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Benenson, R., Omran, M., Hosang, J. H., & Schiele, B. (2014). Ten years of pedestrian detection, what have we learned? In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Bertozzi, M., Bombini, L., Broggi, A., Buzzoni, M., Cardarelli, E., Cattani, S., Cerri, P., Coati, A., Debattisti, S., Falzoni, A., Fedriga, R. I., Felisa, M., Gatti, L., Giacomazzo, A., Grisleri, P., Laghi, M. C., Mazzei, L., Medici, P., Panciroli, M., Porta, P. P., Zani, P., & Versari, P. (2011). VIAC: an out of ordinary experiment. In *Proc. IEEE Intelligent Vehicles Symposium (IV)* (pp. 175–180).

Bertozzi, M., Broggi, A., & Fascioli, A. (2000). Vision-based intelligent vehicles: State of the art and perspectives. *Robotics and Autonomous Systems (RAS)*, *32*, 1–16.

Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2016). Simple online and realtime tracking. In *Proc. IEEE International Conf. on Image Processing (ICIP)*.

Bhotika, R., Fleet, D. J., & Kutulakos, K. N. (2002). A probabilistic theory of occupancy and emptiness. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Black, M. J., & Anandan, P. (1993). A framework for the robust estimation of optical flow. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Blaha, M., Vogel, C., Richard, A., Wegner, J. D., Pock, T., & Schindler, K. (2016). Large-scale semantic 3d reconstruction: An adaptive multi-resolution model for multi-class volumetric labeling. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Bogo, F., Kanazawa, A., Lassner, C., Gehler, P. V., Romero, J., & Black, M. J. (2016). Keep it SMPL: automatic estimation of 3d human pose and shape from a single image. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Bojarski, M., Testa, D. D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U., Zhang, J., Zhang, X., Zhao, J., & Zieba, K. (2016). End to end learning for self-driving cars. *arXiv.org*, *1604.07316*.

yves Bouguet, J. (2000). *Pyramidal implementation of the Lucas Kanade feature tracker*. Technical Report Intel Corporation Microprocessor Research Labs.

Bradski, G., & Kaehler, A. (2008). *Learning OpenCV: Computer Vision with the OpenCV Library*. Cambridge, MA: O'Reilly.

Braid, D., Broggi, A., & Schmiedel, G. (2006). The terramax autonomous vehicle. *Journal of Field Robotics (JFR)*, .

Bredies, K., Kunisch, K., & Pock, T. (2010). Total generalized variation. *Journal of Imaging Sciences (SIAM)*, *3*, 492–526.

Breitenstein, M. D., Reichlin, F., Leibe, B., Koller-Meier, E., & Gool, L. J. V. (2011). Online multiperson tracking-by-detection from a single, uncalibrated camera. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *33*, 1820–1833.

Broggi, A., Bertozzi, M., Fascioli, A., & Conte, G. (1999). *Automatic Vehicle Guidance: the Experience of the ARGO Vehicle*. Singapore: World Scientific.

Broggi, A., Bertozzi, M., Fascioli, A., & Sechi, M. (2000). Shape-based pedestrian detection. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Broggi, A., Cerri, P., Debattisti, S., Laghi, M. C., Medici, P., Molinari, D., Panciroli, M., & Prioletti, A. (2015). PROUD - public road urban driverless-car test. *IEEE Trans. on Intelligent Transportation Systems (TITS)*, *16*, 3508–3519.

Brox, T., Bruhn, A., Papenberg, N., & Weickert, J. (2004). High accuracy optical flow estimation based on a theory for warping. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Brox, T., & Malik, J. (2011). Large displacement optical flow: Descriptor matching in variational motion estimation. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *33*, 500–513.

Brubaker, M. A., Geiger, A., & Urtasun, R. (2016). Map-based probabilistic visual self-localization. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *38*, 652–665.

Bruhn, A., & Weickert, J. (2006). A confidence measure for variational optic flow methods. In R. Klette, R. Kozera, L. Noakes, & J. Weickert (Eds.), *Geometric Properties for Incomplete Data* (pp. 283–298). Dordrecht: Springer Netherlands. doi:10.1007/1-4020-3858-8_15.

Buczko, M., & Willert, V. (2016a). Flow-decoupled normalized reprojection error for visual odometry. In *Proc. IEEE Conf. on Intelligent Transportation Systems (ITSC)*.

Buczko, M., & Willert, V. (2016b). How to distinguish inliers from outliers in visual odometry for high-speed automotive applications. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Budvytis, I., Badrinarayanan, V., & Cipolla, R. (2010). Label propagation in complex video sequences using semi-supervised learning. In *Proc. of the British Machine Vision Conf. (BMVC)*.

Buehler, M., Iagnemma, K., & Singh, S. (2007). *The 2005 darpa grand challenge: The great robot race* volume 36. Springer.

Buehler, M., Iagnemma, K., & Singh, S. (2009). The DARPA urban challenge. *DARPA Challenge*, *56*.

Butler, D. J., Wulff, J., Stanley, G. B., & Black, M. J. (2012). A naturalistic open source movie for optical flow evaluation. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Bódis-Szomorú, A., Riemenschneider, H., & Gool, L. V. (2016). Efficient volumetric fusion of airborne and street-side data for urban reconstruction. In

Proc. of the International Conf. on Pattern Recognition (ICPR).

Cai, Z., Fan, Q., Feris, R. S., & Vasconcelos, N. (2016). A unified multi-scale deep convolutional neural network for fast object detection. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Cech, J., Sanchez-Riera, J., & Horaud, R. P. (2011). Scene flow estimation by growing correspondence seeds. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Chen, C., Seff, A., Kornhauser, A. L., & Xiao, J. (2015a). Deepdriving: Learning affordance for direct perception in autonomous driving. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)* (pp. 2722–2730).

Chen, L.-C., Fidler, S., Yuille, A. L., & Urtasun, R. (2014). Beat the mturkers: Automatic image labeling from weak 3d supervision. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2015b). Semantic image segmentation with deep convolutional nets and fully connected crfs. In *Proc. of the International Conf. on Learning Representations (ICLR)*.

Chen, Q., & Koltun, V. (2016). Full flow: Optical flow estimation by global optimization over regular grids. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Chen, X., Kundu, K., Zhang, Z., Ma, H., Fidler, S., & Urtasun, R. (2016a). Monocular 3d object detection for autonomous driving. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Chen, X., Kundu, K., Zhu, Y., Berneshawi, A. G., Ma, H., Fidler, S., & Urtasun, R. (2015c). 3d object proposals for accurate object class detection. In *Advances in Neural Information Processing Systems (NIPS)*.

Chen, X., Kundu, K., Zhu, Y., Ma, H., Fidler, S., & Urtasun, R. (2016b). 3d object proposals using stereo imagery for accurate object class detection. *arXiv.org, 1608.07711*.

Chen, X., Ma, H., Wan, J., Li, B., & Xia, T. (2016c). Multi-view 3d object detection network for autonomous driving. *arXiv.org, 1611.07759*.

Cherabier, I., Häne, C., Oswald, M. R., & Pollefeys, M. (2016). Multi-label semantic 3d reconstruction using voxel blocks. In *Proc. of the International Conf. on 3D Vision (3DV)*.

Choi, W. (2015). Near-online multi-target tracking with aggregated local flow descriptor. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Choi, W., Pantofaru, C., & Savarese, S. (2013). A general framework for tracking multiple people from a moving camera. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 35, 1577–1591.

Collins, R. T. (1996). A space-sweep approach to true multi-image matching. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (pp. 358–363).

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., & Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Cordts, M., Schneider, L., Enzweiler, M., Franke, U., & Roth, S. (2014). Object-level priors for stixel generation. In *Proc. of the German Conference on Pattern Recognition (GCPR)*.

Cornelis, N., Leibe, B., Cornelis, K., & Van Gool, L. J. (2008). 3D urban scene modeling integrating recognition and reconstruction. *International Journal of Computer Vision (IJCV)*, 78, 121–141.

Cummins, M., & Newman, P. (2008). Fab-map: Probabilistic localization and mapping in the space of appearance. *International Journal of Robotics Research (IJRR)*, 27, 647–665.

Curless, B., & Levoy, M. (1996). A volumetric method for building complex models from range images. In *ACM Trans. on Graphics (SIGGRAPH)*.

Cvisic, I., & Petrovic, I. (2015). Stereo odometry based on careful feature selection and tracking. In *Proc. European Conf. on Mobile Robotics (ECMR)*.

Dai, J., He, K., & Sun, J. (2016). Instance-aware semantic segmentation via multi-task network cascades. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Dame, A., Prisacariu, V., Ren, C., & Reid, I. (2013). Dense reconstruction using 3D object shape priors. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Deigmoeller, J., & Eggert, J. (2016). Stereo visual odometry without temporal filtering. In *Proc. of the German Conference on Pattern Recognition (GCPR)*.

Dellaert, F., Fox, D., Burgard, W., & Thrun, S. (1999). Monte carlo localization for mobile robots. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Dellaert, F., & Kaess, M. (2006). Square root sam: Simultaneous localization and mapping via square root information smoothing. *International Journal of Robotics Research (IJRR)*, .

Deng, J., Dong, W., Socher, R., jia Li, L., Li, K., & Fei-fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Derome, M., Plyer, A., Sanfourche, M., & Le Besnerais, G. (2016). A prediction-correction approach for real-time optical flow computation using stereo. In *Proc. of the German Conference on Pattern Recognition (GCPR)*.

Desai, C., Ramanan, D., & Fowlkes, C. C. (2011). Discriminative models for multi-class object layout. *International Journal of Computer Vision (IJCV)*, 95, 1–12.

Dickmanns, E. D., Behringer, R., Dickmanns, D., Hildebrandt, T., Maurer, M., Thomanek, F., & Schiehlen, J. (1994). The seeing passenger car 'vamors-p'. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Dickmanns, E. D., & Graefe, V. (1988). Dynamic monocular machine vision. *Machine Vision and Applications (MVA)*, 1, 223–240.

Dickmanns, E. D., & Mysliwetz, B. D. (1992). Recursive 3-d road and relative ego-state recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 14, 199–213.

Dickmanns, E. D., Mysliwetz, B. D., & Christians, T. (1990). An integrated spatio-temporal approach to automatic visual guidance of autonomous vehicles. *IEEE Trans. on Systems, Man and Cybernetics (TSMC)*, 20, 1273–1284.

Dicle, C., Camps, O. I., & Sznaier, M. (2013). The way they move: Tracking multiple targets with similar appearance. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Dollár, P., Appel, R., Belongie, S. J., & Perona, P. (2014). Fast feature pyramids for object detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 36, 1532–1545.

Dollar, P., Wojek, C., Schiele, B., & Perona, P. (2009). Pedestrian detection: A benchmark. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Dollar, P., Wojek, C., Schiele, B., & Perona, P. (2011). Pedestrian detection: An evaluation of the state of the art. In *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*. volume 99.

Dollár, P., Wojek, C., Schiele, B., & Perona, P. (2012). Pedestrian detection: An evaluation of the state of the art. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 34, 743–761.

Dosovitskiy, A., Fischer, P., Ilg, E., Haeusser, P., Hazirbas, C., Golkov, V., v.d. Smagt, P., Cremers, D., & Brox, T. (2015). Flownet: Learning optical flow with convolutional networks. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Drory, A., Haubold, C., Avidan, S., & Hamprecht, F. A. (2014). Semi-global matching: A principled derivation in terms of message passing. In *Proc. of the German Conference on Pattern Recognition (GCPR)*.

Duan, L., & Lafarge, F. (2016). Towards large-scale city reconstruction from satellites. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Dubé, R., Dugas, D., Stumm, E., Nieto, J. I., Siegwart, R., & Cadena, C. (2016). Segmatch: Segment based loop-closure for 3d point clouds. *arXiv.org, 1609.07720*.

Einecke, N., & Eggert, J. (2013). Stereo image warping for improved depth estimation of road surfaces. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Einecke, N., & Eggert, J. (2014). Block-matching stereo with relaxed frontoparallel assumption. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Engel, J., Koltun, V., & Cremers, D. (2016). Direct sparse odometry. *arXiv.org, 1607.02565*.

Engel, J., Schöps, T., & Cremers, D. (2014). LSD-SLAM: large-scale direct monocular SLAM. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Engel, J., Stückler, J., & Cremers, D. (2015). Large-scale direct SLAM with stereo cameras. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*.

Engel, J., Sturm, J., & Cremers, D. (2013). Semi-dense visual odometry for a monocular camera. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Engelcke, M., Rao, D., Wang, D. Z., Tong, C. H., & Posner, I. (2016). Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks. *arXiv.org*, *609.06666*.

Enzweiler, M., & Gavrila, D. (2008). A mixed generative-discriminative framework for pedestrian classification. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Enzweiler, M., & Gavrila, D. M. (2009). Monocular pedestrian detection: Survey and experiments. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *31*, 2179–2195.

Enzweiler, M., & Gavrila, D. M. (2011). A multilevel mixture-of-experts framework for pedestrian classification. *IEEE Trans. on Image Processing (TIP)*, *20*, 2967–2979.

Erbs, F., Schwarz, B., & Franke, U. (2012). Stixmentation - probabilistic stixel based traffic scene labeling. In *Proc. of the British Machine Vision Conf. (BMVC)*.

Erbs, F., Schwarz, B., & Franke, U. (2013). From stixels to objects - A conditional random field based approach. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Ess, A., Leibe, B., Schindler, K., & Gool, L. V. (2009a). Robust multi-person tracking from a mobile platform. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *31*, 1831–1846.

Ess, A., Leibe, B., Schindler, K., & Van Gool, L. (2008). A mobile vision system for robust multi-person tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Ess, A., Mueller, T., Grabner, H., & van Gool, L. (2009b). Segmentation-based urban traffic scene understanding. In *Proc. of the British Machine Vision Conf. (BMVC)*.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision (IJCV)*, *88*, 303–338.

Fagot-Bouquet, L., Audigier, R., Dhome, Y., & Lerasle, F. (2016). Improving multi-frame data association with sparse representations for robust near-online multi-object tracking. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Farneback, G. (2003). Two-frame motion estimation based on polynomial expansion. In *Scandinavian Conference on Image Analysis (SCIA)*.

Faugeras, O. D., & Keriven, R. (1998). Variational principles, surface evolution, pdes, level set methods, and the stereo problem. *IEEE Trans. on Image Processing (TIP)*, *7*, 336–344.

Felzenszwalb, P., Girshick, R., McAllester, D., & Ramanan, D. (2010). Object detection with discriminatively trained part based models. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *32*, 1627–1645.

Felzenszwalb, P. F., McAllester, D. A., & Ramanan, D. (2008). A discriminatively trained, multiscale, deformable part model. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Ferryman, J., & Shahrokni, A. (2009). Pets2009: Dataset and challenge. In *Performance Evaluation of Tracking and Surveillance* (pp. 1–6).

Floros, G., & Leibe, B. (2012). Joint 2d-3d temporally consistent semantic segmentation of street scenes. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Frahm, J.-M., Fite-Georgel, P., Gallup, D., Johnson, T., Raguram, R., Wu, C., Jen, Y.-H., Dunn, E., Clipp, B., Lazebnik, S., & Pollefeys, M. (2010). Building rome on a cloudless day. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Franke, U., Gavrila, D., Görzig, S., Lindner, F., Paetzold, F., & Wöhler, C. (1998). Autonomous driving goes downtown. *Intelligent Systems (IS)*, *13*, 40–48.

Franke, U., & Joos, A. (2000). Real-time stereo vision for urban traffic scene understanding. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Franke, U., Mehring, S., Suissa, A., & Hahn, S. (1994). The daimler-benz steering assistant: a spin-off from autonomous driving. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Franke, U., Rabe, C., Badino, H., & Gehrig, S. (2005). 6D-Vision: fusion of stereo and motion for robust environment perception. In *Proc. of the DAGM Symposium on Pattern Recognition (DAGM)*.

Fraundorfer, F., & Scaramuzza, D. (2011). Visual odometry: Part ii - matching, robustness, and applications. *Robotics and Automation Magazine (RAM)*, .

Fritsch, J., Kuehnl, T., & Geiger, A. (2013). A new performance measure and evaluation benchmark for road detection algorithms. In *Proc. IEEE Conf. on Intelligent Transportation Systems (ITSC)*.

Frost, D. P., Kähler, O., & Murray, D. W. (2016). Object-aware bundle adjustment for correcting monocular scale drift. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Früh, C., Jain, S., & Zakhor, A. (2005). Data processing algorithms for generating textured 3d building facade meshes from laser scans and camera images. *International Journal of Computer Vision (IJCV)*, *61*, 159–184.

Furgale, P. T., Schwesinger, U., Rufli, M., Derendarz, W., Grimmett, H., Mühlfellner, P., Wonneberger, S., Timpner, J., Rottmann, S., Li, B., Schmidt, B., Nguyen, T., Cardarelli, E., Cattani, S., Bruning, S., Horstmann, S., Stellmacher, M., Mielenz, H., Köser, K., Beermann, M., Hane, C., Heng, L., Lee, G. H., Fraundorfer, F., Iser, R., Triebel, R., Posner, I., Newman, P., Wolf, L. C., Pollefeys, M., Brosig, S., Effertz, J., Pradalier, C., & Siegwart, R. (2013). Toward automated driving in cities using close-to-market sensors: An overview of the v-charge project. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Furukawa, Y., & Ponce, J. (2010). Accurate, dense, and robust multi-view stereopsis. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *32*, 1362–1376.

Gadde, R., Jampani, V., Kiefel, M., Kappler, D., & Gehler, P. V. (2016a). Superpixel convolutional networks using bilateral inceptions. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Gadde, R., Jampani, V., Marlet, R., & Gehler, P. V. (2016b). Efficient 2d and 3d facade segmentation using auto-context. *arXiv.org*, *1606.06437*.

Gadot, D., & Wolf, L. (2016). Patchbatch: a batch augmented loss for optical flow. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Gaidon, A., & Vig, E. (2015). Online domain adaptation for multi-object tracking. In *Proc. of the British Machine Vision Conf. (BMVC)*.

Gaidon, A., Wang, Q., Cabon, Y., & Vig, E. (2016). Virtual worlds as proxy for multi-object tracking analysis. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Gallup, D., Frahm, J. M., Mordohai, P., & Pollefeys, M. (2008). Variable baseline/resolution stereo. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Gallup, D., Frahm, J.-M., & Pollefeys, M. (2010). Piecewise planar and non-planar stereo for urban scene reconstruction. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Gálvez-López, D., & Tardós, J. D. (2012). Bags of binary words for fast place recognition in image sequences. *IEEE Trans. on Robotics*, *28*, 1188–1197.

Gavrila, D. M., & Munder, S. (2007). Multi-cue pedestrian detection and tracking from a moving vehicle. *International Journal of Computer Vision (IJCV)*, *73*, 41–59.

Gehrig, S. K., Eberli, F., & Meyer, T. (2009). A real-time low-power stereo vision engine using semi-global matching. In *Proc. of the International Conf. on Computer Vision Systems (ICVS)*.

Geiger, A. (2009). Monocular road mosaicing for urban environments. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Geiger, A., Lauer, M., Moosmann, F., Ranft, B., Rapp, H., Stiller, C., & Ziegler, J. (2012a). Team annieway's entry to the grand cooperative driving challenge 2011. *IEEE Trans. on Intelligent Transportation Systems (TITS)*, *13*, 1008–1017.

Geiger, A., Lauer, M., Wojek, C., Stiller, C., & Urtasun, R. (2014). 3D traffic scene understanding from movable platforms. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *36*, 1012–1025.

Geiger, A., Lenz, P., Stiller, C., & Urtasun, R. (2013). Vision meets robotics: The KITTI dataset. *International Journal of Robotics Research (IJRR)*, *32*, 1231–1237.

Geiger, A., Lenz, P., & Urtasun, R. (2012b). Are we ready for autonomous driving? The KITTI vision benchmark suite. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Geiger, A., Moosmann, F., Car, O., & Schuster, B. (2012c). Automatic calibration of range and camera sensors using a single shot. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Geiger, A., Roser, M., & Urtasun, R. (2010). Efficient large-scale stereo matching. In *Proc. of the Asian Conf. on Computer Vision (ACCV)*.

Geiger, A., Ziegler, J., & Stiller, C. (2011). StereoScan: Dense 3D reconstruction in real-time. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Gerke, M. (2015). *Use of the stair vision library within the ISPRS 2D semantic labelingbenchmark (Vaihingen)*. Technical Report University of Twente.

Geronimo, D., Lopez, A. M., Sappa, A. D., & Graf, T. (2010). Survey on pedestrian detection for advanced driver assistance systems. *IEEE Trans. on*

Pattern Analysis and Machine Intelligence (PAMI), *32*, 1239–1258.

Geyer, C., & Daniilidis, K. (2000). A unifying theory for central panoramic systems and practical implications. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Ghiasi, G., & Fowlkes, C. C. (2016). Laplacian pyramid reconstruction and refinement for semantic segmentation. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Giebel, J., Gavrila, D., & Schnörr, C. (2004). A bayesian framework for multi-cue 3d object tracking. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Girshick, R. B. (2015). Fast R-CNN. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Gkioxari, G., Hariharan, B., Girshick, R., & Malik, J. (2014). Using k-poselets for detecting people and localizing their keypoints. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

González, A., Villalonga, G., Xu, J., Vázquez, D., Amores, J., & López, A. M. (2015). Multiview random forest of local experts combining RGB and LIDAR data for pedestrian detection. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

González, A., Vázquez, D., Lóopez, A. M., & Amores, J. (2016). On-board object detection: Multicue, multimodal, and multiview random forest of local experts. *IEEE Trans. on Cybernetics*, .

Grimmett, H., Bürki, M., Paz, L. M., Pinies, P., Furgale, P. T., Posner, I., & Newman, P. (2015). Integrating metric and semantic maps for vision-only automated parking. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Grisleri, P., & Fedriga, I. (2010). The braive platform. In *IFAC*.

Günyel, B., Benenson, R., Timofte, R., & Gool, L. J. V. (2012). Stixels motion estimation without optical flow computation. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Güney, F., & Geiger, A. (2015). Displets: Resolving stereo ambiguities using object knowledge. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Güney, F., & Geiger, A. (2016). Deep discrete flow. In *Proc. of the Asian Conf. on Computer Vision (ACCV)*.

Hackel, T., Wegner, J. D., & Schindler, K. (2016). Fast semantic segmentation of 3d point clouds with strongly varying density. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences (APRS)*, *III-3*, 177 – 184.

Haene, C., Savinov, N., & Pollefeys, M. (2014). Class specific 3d object shape priors using surface normals. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Haene, C., Zach, C., Cohen, A., Angst, R., & Pollefeys, M. (2013). Joint 3D scene reconstruction and class segmentation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Haene, C., Zach, C., Zeisl, B., & Pollefeys, M. (2012). A patch prior for dense 3d reconstruction in man-made environments. In *Proc. of the International Conf. on 3D Digital Imaging, Modeling, Data Processing, Visualization and Transmission (THREEDIMPVT)*.

Häne, C., Heng, L., Lee, G. H., Sizov, A., & Pollefeys, M. (2014). Real-time direct dense matching on fisheye images using plane-sweeping stereo. In *Proc. of the International Conf. on 3D Vision (3DV)*.

Häne, C., Sattler, T., & Pollefeys, M. (2015). Obstacle detection for self-driving cars using only monocular cameras and wheel odometry. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*.

Hayder, Z., He, X., & Salzmann, M. (2016). Shape-aware instance segmentation. *arXiv.org*, *1612.03129*.

Hays, J., & Efros, A. A. (2008). im2gps: estimating geographic information from a single image. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

He, K., Zhang, X., Ren, S., & Sun, J. (2014). Spatial pyramid pooling in deep convolutional networks for visual recognition. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

He, X., Zemel, R. S., & Carreira-Perpinan, M. A. (2004). Multiscale conditional random fields for image labeling. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

He, X., Zemel, R. S., & Ray, D. (2006). Learning and incorporating top-down cues in image segmentation. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Heeger, D. J. (1988). Optical flow using spatiotemporal filters. *International Journal of Computer Vision (IJCV)*, *1*, 279–302.

Heng, L., Furgale, P. T., & Pollefeys, M. (2015). Leveraging image-based localization for infrastructure-based calibration of a multi-camera rig. *Journal of Field Robotics (JFR)*, *32*, 775–802.

Heng, L., Li, B., & Pollefeys, M. (2013). Camodocal: Automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*.

Hirschmüller, H. (2008). Stereo processing by semiglobal matching and mutual information. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *30*, 328–341.

Hirschmüller, H., & Scharstein, D. (2007). Evaluation of cost functions for stereo matching. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (pp. 1–8).

Hoiem, D., Efros, A., & Hebert, M. (2008). Putting objects in perspective. *International Journal of Computer Vision (IJCV)*, *80*, 3–15.

Hoiem, D., Efros, A. A., & Hebert, M. (2007). Recovering surface layout from an image. *International Journal of Computer Vision (IJCV)*, *75*, 151–172.

Honauer, K., Maier-Hein, L., & Kondermann, D. (2015). The hci stereo metrics: Geometry-aware performance analysis of stereo algorithms. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Horn, B. K. P., & Schunck, B. G. (1981). Determining optical flow. *Artificial Intelligence (AI)*, *17*, 185–203.

Hu, Y., Song, R., & Li, Y. (2016). Efficient coarse-to-fine patchmatch for large displacement optical flow. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Huang, J., & You, S. (2016). Point cloud labeling using 3d convolutional neural network. In *Proc. of the International Conf. on Pattern Recognition (ICPR)*.

Huguet, F., & Devernay, F. (2007). A variational method for scene flow estimation from stereo sequences. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Hur, J., & Roth, S. (2016). Joint optical flow and temporally consistent semantic segmentation. In *Proc. of the European Conf. on Computer Vision (ECCV) Workshops*.

Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., & Brox, T. (2016). Flownet 2.0: Evolution of optical flow estimation with deep networks. *arXiv.org*, *1612.01925*.

Jampani, V., Kiefel, M., & Gehler, P. V. (2016). Learning sparse high dimensional filters: Image filtering, dense crfs and bilateral neural networks. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Janai, J., Güney, F., Wulff, J., Black, M., & Geiger, A. (2017). Slow flow: Exploiting high-speed cameras for accurate and diverse optical flow reference data. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Jensen, R. R., Dahl, A. L., Vogiatzis, G., Tola, E., & Aanæs, H. (2014). Large scale multi-view stereopsis evaluation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R. B., Guadarrama, S., & Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. In *Proc. of the International Conf. on Multimedia (ICM)*.

Julier, S. J., & Uhlmann, J. K. (2001). A counter example to the theory of simultaneous localization and map building. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J. J., & Dellaert, F. (2012). iSAM2: Incremental smoothing and mapping using the Bayes tree. *International Journal of Robotics Research (IJRR)*, *31*, 217–236.

Kaess, M., Ni, K., & Dellaert, F. (2009). Flow separation for fast and robust stereo odometry. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Kaess, M., Ranganathan, A., & Dellaert, F. (2008). iSAM: Incremental smoothing and mapping. *IEEE Trans. on Robotics*, *24*, 1365–1378.

Kaminsky, R. S., Snavely, N., Seitz, S. M., & Szeliski, R. (2009). Alignment of 3d point clouds to overhead images. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Kendall, A., Grimes, M., & Cipolla, R. (2015). Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proc. of the IEEE Inter-*

national Conf. on Computer Vision (ICCV).

Kieritz, H., Becker, S., Hubner, W., & Arens, M. (2016). Online multi-person tracking using integral channel features. In *Proc. of International Conf. on Advanced Video and Signal Based Surveillance (AVSS).*

Kim, C., Li, F., Ciptadi, A., & Rehg, J. M. (2015). Multiple hypothesis tracking revisited. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV).*

Kirillov, A., Levinkov, E., Andres, B., Savchynskyy, B., & Rother, C. (2016). Instancecut: from edges to instances with multicut. *arXiv.org, 1611.08272.*

Kitt, B., Geiger, A., & Lategahn, H. (2010). Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme. In *Proc. IEEE Intelligent Vehicles Symposium (IV).*

Klette, R. (2015). *Vision-based Driver Assistance Systems.* Technical Report CITR, Auckland, New Zealand.

Kohli, P., Ladicky, L., & Torr, P. H. S. (2009). Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision (IJCV), 82,* 302–324.

Kolmogorov, V. (2006). Convergent tree-reweighted message passing for energy minimization. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 28,* 1568–1583.

Kondermann, C., Kondermann, D., Jähne, B., & Garbe, C. S. (2007). An adaptive confidence measure for optical flows based on linear subspace projections. In *Proc. of the German Conference on Pattern Recognition (GCPR)* (pp. 132–141).

Kondermann, C., Mester, R., & Garbe, C. S. (2008). A statistical confidence measure for optical flows. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Kondermann, D., Nair, R., Honauer, K., Krispin, K., Andrulis, J., Brock, A., Gussefeld, B., Rahimimoghaddam, M., Hofmann, S., Brenner, C., & Jahne, B. (2016). The hci benchmark suite: Stereo and flow ground truth with uncertainties for urban autonomous driving. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) Workshops.*

Krähenbühl, P., & Koltun, V. (2011). Efficient inference in fully connected CRFs with Gaussian edge potentials. In *Advances in Neural Information Processing Systems (NIPS).*

Krešo, I., Čaušević, D., Krapac, J., & Šegvić, S. (2016). Convolutional scale invariance for semantic segmentation. In *Proc. of the German Conference on Pattern Recognition (GCPR).*

Krešo, I., & Šegvić, S. (2015). Improving the egomotion estimation by correcting the calibration bias. In *Proc. of the Conf. on Computer Vision Theory and Applications (VISAPP).*

Kroeger, T., Timofte, R., Dai, D., & Gool, L. V. (2016). Fast optical flow using dense inverse search. *arXiv.org, 1603.03590.*

Kuehnl, T., Kummert, F., & Fritsch, J. (2012). Spatial ray features for real-time ego-lane extraction. In *Proc. IEEE Intelligent Vehicles Symposium (IV).*

Kuettel, D., Breitenstein, M. D., Gool, L. V., & Ferrari, V. (2010). What's going on?: Discovering spatio-temporal dependencies in dynamic scenes. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Kumar, S., & Hebert, M. (2005). A hierarchical field framework for unified context-based classification. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV).*

Kümmerle, R., Grisetti, G., Strasdat, H., Konolige, K., & Burgard, W. (2011). G^{2}o: A general framework for graph optimization. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)* (pp. 3607–3613).

Kundu, A., Li, Y., Dellaert, F., Li, F., & Rehg, J. (2014). Joint semantic segmentation and 3d reconstruction from monocular video. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Kundu, A., Vineet, V., & Koltun, V. (2016). Feature space optimization for semantic video segmentation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Kuschk, G., & Cremers, D. (2013). Fast and accurate large-scale stereo reconstruction using variational methods. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV) Workshops.*

Kutulakos, K. N., & Seitz, S. M. (2000). A theory of shape by space carving. *International Journal of Computer Vision (IJCV), 38,* 199–218.

Kybic, J., & Nieuwenhuis, C. (2011). Bootstrap optical flow confidence and uncertainty measure. *Computer Vision and Image Understanding (CVIU), 115,* 1449–1462.

Labatut, P., Pons, J., & Keriven, R. (2007). Efficient multi-view reconstruction of large-scale scenes using interest points, delaunay triangulation and graph cuts. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*

(pp. 1–8).

Laddha, A., Kocamaz, M. K., Navarro-Serment, L. E., & Hebert, M. (2016). Map-supervised road detection. In *Proc. IEEE Intelligent Vehicles Symposium (IV).*

Ladicky, L., Russell, C., Kohli, P., & Torr, P. (2009). Associative hierarchical crfs for object class image segmentation. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV).*

Ladicky, L., Russell, C., Kohli, P., & Torr, P. H. (2010). Graph cut based inference with co-occurrence statistics. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Ladicky, L., Russell, C., Kohli, P., & Torr, P. H. S. (2014). Associative hierarchical random fields. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 36,* 1056–1077.

Lafarge, F., Descombes, X., Zerubia, J., & Deseilligny, M. P. (2010). Structural approach for building reconstruction from a single DSM. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 32,* 135–147.

Lafarge, F., Keriven, R., Bredif, M., & Vu, H.-H. (2013). A hybrid multiview stereo algorithm for modeling urban scenes. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 35,* 5–17.

Lafarge, F., & Mallet, C. (2012). Creating large-scale city models from 3d-point clouds: A robust approach with hybrid representation. *International Journal of Computer Vision (IJCV), 99,* 69–85.

Lategahn, H., Geiger, A., & Kitt, B. (2011). Visual slam for autonomous ground vehicles. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA).*

Le, N., Heili, A., & Odobez, J.-M. (2016). Long-term time-sensitive costs for crf-based tracking by detection. In *Proc. of the European Conf. on Computer Vision (ECCV) Workshops.*

Leal-Taixé, L., Milan, A., Reid, I. D., Roth, S., & Schindler, K. (2015). Motchallenge 2015: Towards a benchmark for multi-target tracking. *arXiv.org, 1504.01942.*

Lee, B., Erdenee, E., Jin, S., Nam, M. Y., Jung, Y. G., & Rhee, P. (2016a). Multi-class multi-object tracking using changing point detection. In *Proc. of the European Conf. on Computer Vision (ECCV) Workshops.*

Lee, B., Erdenee, E., Jin, S., Nam, M. Y., Jung, Y. G., & Rhee, P. (2016b). Multi-class multi-object tracking using changing point detection. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Lee, G. H., Fraundorfer, F., & Pollefeys, M. (2013a). Motion estimation for self-driving cars with a generalized camera. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Lee, G. H., Fraundorfer, F., & Pollefeys, M. (2013b). Structureless pose-graph loop-closure with a multi-camera system on a self-driving car. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS).*

Lee, G. H., Pollefeys, M., & Fraundorfer, F. (2014). Relative pose estimation for a multi-camera system with known vertical direction. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Leibe, B., Cornelis, N., Cornelis, K., & Van Gool, L. (2007). Dynamic 3d scene analysis from a moving vehicle. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Leibe, B., Leonardis, A., & Schiele, B. (2008a). Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision (IJCV), 77,* 259–289.

Leibe, B., Schindler, K., Cornelis, N., & Van Gool, L. (2008b). Coupled detection and tracking from static cameras and moving vehicles. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 30,* 1683–1698.

Lenz, P., Geiger, A., & Urtasun, R. (2015). Followme: Efficient online min-cost flow tracking with bounded memory and computation. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV).*

Leutenegger, S., Furgale, P. T., Rabaud, V., Chli, M., Konolige, K., & Siegwart, R. (2013). Keyframe-based visual-inertial SLAM using nonlinear optimization. In *Proc. Robotics: Science and Systems (RSS).*

Levi, D., Garnett, N., & Fetaya, E. (2015). Stixelnet: A deep convolutional network for obstacle detection and road segmentation. In *Proc. of the British Machine Vision Conf. (BMVC).*

Levinkov, E., Tang, S., Insafutdinov, E., & Andres, B. (2016). Joint graph decomposition and node labeling by local search. *arXiv.org, 1611.04399.*

Levinson, J., Montemerlo, M., & Thrun, S. (2007). Map-based precision vehicle localization in urban environments. In *Proc. Robotics: Science and Systems (RSS).*

Levinson, J., & Thrun, S. (2010). Robust vehicle localization in urban environments using probabilistic maps. In *Proc. IEEE International Conf. on*

Robotics and Automation (ICRA).

Li, A., Chen, D., Liu, Y., & Yuan, Z. (2016a). Coordinating multiple disparity proposals for stereo computation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Li, B., Wu, T., & Zhu, S.-C. (2014). Integrating context and occlusion for car detection by hierarchical and-or model. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Li, B., Zhang, T., & Xia, T. (2016b). Vehicle detection from 3d lidar using fully convolutional network. In *Proc. Robotics: Science and Systems (RSS).*

Li, Y., Crandall, D. J., & Huttenlocher, D. P. (2009). Landmark classification in large-scale image collections. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV).*

Li, Y., Min, D., Brown, M. S., Do, M. N., & Lu, J. (2015). Spm-bp: Sped-up patchmatch belief propagation for continuous mrfs. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV).*

Li, Y., Snavely, N., Huttenlocher, D., & Fua, P. (2012). Worldwide pose estimation using 3d point clouds. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Li, Z., & Chen, J. (2015). Superpixel segmentation using linear spectral clustering. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (pp. 1356–1363).

Lin, G., Milan, A., Shen, C., & Reid, I. D. (2016a). Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. *arXiv.org, 1611.06612.*

Lin, G., Shen, C., Reid, I. D., & van den Hengel, A. (2016b). Efficient piecewise training of deep structured models for semantic segmentation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Lin, T., Belongie, S. J., & Hays, J. (2013). Cross-view image geolocalization. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (pp. 891–898).

Lin, T., Cui, Y., Belongie, S. J., & Hays, J. (2015). Learning deep representations for ground-to-aerial geolocalization. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Liu, X., Zhao, Y., & Zhu, S.-C. (2014). Single-view 3D scene parsing by attributed grammar. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Liu, Z., Li, X., Luo, P., Loy, C. C., & Tang, X. (2015). Semantic image segmentation via deep parsing network. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV).*

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Longuet-Higgins, H. (1981). A computer algorithm for reconstructing a scene from two projections. *Nature, 293,* 133–135.

Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., & Black, M. J. (2015). SMPL: A skinned multi-person linear model. *ACM Trans. on Graphics (SIGGRAPH),* .

Lorensen, W. E., & Cline, H. E. (1987). Marching cubes: A high resolution 3d surface construction algorithm. In *ACM Trans. on Graphics (SIGGRAPH).*

Lowry, S. M., Sünderhauf, N., Newman, P., Leonard, J. J., Cox, D. D., Corke, P. I., & Milford, M. J. (2016). Visual place recognition: A survey. *IEEE Trans. on Robotics, 32,* 1–19.

Luo, W., Schwing, A., & Urtasun, R. (2016). Efficient deep learning for stereo matching. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Lv, Z., Beall, C., Alcantarilla, P., Li, F., Kira, Z., & Dellaert, F. (2016). A continuous optimization approach for efficient and accurate scene flow. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Mac Aodha, O., Humayun, A., Pollefeys, M., & Brostow, G. J. (2013). Learning a confidence measure for optical flow. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 35,* 1107–1120.

Maddern, W., Pascoe, G., Linegar, C., & Newman, P. (2016). 1 year, 1000km: The oxford robotcar dataset. *International Journal of Robotics Research (IJRR),* .

Maggiori, E., Tarabalka, Y., & andP Pierre Alliez, G. C. (2016). High-resolution semantic labeling with convolutional neural networks. *arXiv.org, 1611.01962.*

Mansinghka, V., Kulkarni, T., Perov, Y., & Tenenbaum, J. (2013). Approximate bayesian image interpretation using generative probabilistic graphics programs. In *Advances in Neural Information Processing Systems (NIPS).*

Marmanis, D., Schindler, K., Wegner, J. D., Galliani, S., Datcu, M., & Stilla, U. (2016a). Classification with an edge: Improving semantic image segmentation with boundary detection. *arXiv.org, 1612.01337.*

Marmanis, D., Wegner, J. D., Galliani, S., Schindler, K., Datcu, M., & Stilla, U. (2016b). Semantic segmentation of aerial images with an ensemble of cnns. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences (APRS),* (pp. 473–480).

Martinović, A., Knopp, J., Riemenschneider, H., & Van Gool, L. (2015). 3d all the way: Semantic segmentation of urban scenes from start to end in 3d. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Mathias, M., Martinovic, A., & Gool, L. V. (2016). ATLAS: A three-layered approach to facade parsing. *International Journal of Computer Vision (IJCV), 118,* 22–48.

Mattyus, G., Wang, S., Fidler, S., & Urtasun, R. (2015). Enhancing road maps by parsing aerial images around the world. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV).*

Mattyus, G., Wang, S., Fidler, S., & Urtasun, R. (2016). Hd maps: Fine-grained road segmentation by parsing ground and aerial images. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Mayer, N., Ilg, E., Haeusser, P., Fischer, P., Cremers, D., Dosovitskiy, A., & Brox, T. (2016). A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR.*

McCormac, J., Handa, A., Davison, A. J., & Leutenegger, S. (2016). Semanticfusion: Dense 3d semantic mapping with convolutional neural networks. *arXiv.org, 1609.05130.*

Mei, C., & Rives, P. (2007). Single view point omnidirectional camera calibration from planar grids. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA).*

Menze, M., & Geiger, A. (2015). Object scene flow for autonomous vehicles. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Menze, M., Heipke, C., & Geiger, A. (2015a). Discrete optimization for optical flow. In *Proc. of the German Conference on Pattern Recognition (GCPR).*

Menze, M., Heipke, C., & Geiger, A. (2015b). Joint 3d estimation of vehicles and scene flow. In *Proc. of the ISPRS Workshop on Image Sequence Analysis (ISA).*

Micusik, B., & Kosecka, J. (2009). Piecewise planar city 3d modeling from street view panoramic sequences. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Milan, A., Leal-Taixé, L., Reid, I. D., Roth, S., & Schindler, K. (2016). MOT16: A benchmark for multi-object tracking. *arXiv.org, 1603.00831.*

Milan, A., Roth, S., & Schindler, K. (2014). Continuous energy minimization for multitarget tracking. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 36,* 58–72.

Milan, A., Schindler, K., & Roth, S. (2013). Detection- and trajectory-level exclusion in multiple object tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Mirabdollah, M. H., & Mertsching, B. (2014). On the second order statistics of essential matrix elements. In *Proc. of the German Conference on Pattern Recognition (GCPR).*

Mirabdollah, M. H., & Mertsching, B. (2015). Fast techniques for monocular visual odometry. In *Proc. of the German Conference on Pattern Recognition (GCPR).*

Mitzel, D., & Leibe, B. (2012). Taking mobile multi-object tracking to the next level: People, unknown objects, and carried items. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Mohan, R. (2014). Deep deconvolutional networks for scene parsing. *arXiv.org, 1411.4101.*

Montemerlo, M., Thrun, S., Koller, D., & Wegbreit, B. (2002). Fastslam: A factored solution to the simultaneous localization and mapping problem. In *Artificial Intelligence (AI).*

Montoya, J., Wegner, J. D., Ladicky, L., & Schindler, K. (2015). Semantic segmentation of aerial images in urban areas with class-specific higher-order cliques. In *In ISPRS Conf. Photogrammetric Image Analysis (PIA).*

Mueggler, E., Forster, C., Baumli, N., Gallego, G., & Scaramuzza, D. (2015a). Lifetime estimation of events from dynamic vision sensors. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA).*

Mueggler, E., Gallego, G., & Scaramuzza, D. (2015b). Continuous-time trajectory estimation for event-based vision sensors. In *Proc. Robotics: Science*

and Systems (RSS).

Munoz, D., Bagnell, J. A., & Hebert, M. (2010). Stacked hierarchical labeling. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Mur-Artal, R., Montiel, J. M. M., & Tardós, J. D. (2015). ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. on Robotics, 31,* 1147–1163.

Musialski, P., Wonka, P., Aliaga, D. G., Wimmer, M., Gool, L. J. V., & Purgathofer, W. (2013). A survey of urban reconstruction. *Computer Graphics Forum, 32,* 146–177.

Nießner, M., Zollhöfer, M., Izadi, S., & Stamminger, M. (2013). Real-time 3d reconstruction at scale using voxel hashing. In *ACM Trans. on Graphics (SIGGRAPH).*

Nistér, D. (2004). An efficient solution to the five-point relative pose problem. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 26,* 756–777.

Oh, S. M., Tariq, S., Walker, B. N., & Dellaert, F. (2004). Map-based priors for localization. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS).*

Ohn-Bar, E., & Trivedi, M. M. (2015). Learning to detect vehicles by clustering appearance patterns. *IEEE Trans. on Intelligent Transportation Systems (TITS), 16,* 2511–2521.

Oliveira, G., Burgard, W., & Brox, T. (2016). Efficient deep methods for monocular road segmentation. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS).*

de Oliveira, V. M., Santos, V., Sappa, A. D., Dias, P., & Moreira, A. P. (2016). Incremental scenario representations for autonomous driving using geometric polygonal primitives. *Robotics and Autonomous Systems (RAS), 83,* 312–325.

Paisitkriangkrai, S., Sherrah, J., Janney, P., & van den Hengel, A. (2015). Effective semantic pixel labelling with convolutional networks and conditional random fields. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) Workshops.*

Papandreou, G., Chen, L.-C., Murphy, K. P., & Yuille, A. L. (2015). Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV).*

Paskin, M. A. (2003). Thin junction tree filters for simultaneous localization and mapping. In *Proc. of the International Joint Conf. on Artificial Intelligence (IJCAI)* (pp. 1157–1166).

Paul, R., & Newman, P. (2010). FAB-MAP 3D: Topological mapping with spatial and visual appearance. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA).*

Pepik, B., Stark, M., Gehler, P., & Schiele, B. (2013). Occlusion patterns for object class detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Pepik, B., Stark, M., Gehler, P. V., & Schiele, B. (2015). Multi-view and 3d deformable part models. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 37,* 2232–2245.

Persson, M., Piccini, T., Felsberg, M., & Mester, R. (2015). Robust stereo visual odometry from monocular techniques. In *Proc. IEEE Intelligent Vehicles Symposium (IV).*

Pfeiffer, D., & Franke, U. (2010). Efficient representation of traffic scenes by means of dynamic stixels. In *Proc. IEEE Intelligent Vehicles Symposium (IV).*

Pfeiffer, D., & Franke, U. (2011). Towards a global optimal multi-layer stixel representation of dense 3d data. In *Proc. of the British Machine Vision Conf. (BMVC).*

Pinggera, P., Franke, U., & Mester, R. (2015). High-performance long range obstacle detection using stereo vision. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS).*

Pinggera, P., Ramos, S., Gehrig, S., Franke, U., Rother, C., & Mester, R. (2016). Lost and found: detecting small road hazards for self-driving vehicles. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS).*

Piramanayagam, S., Schwartzkopf, W., Koehler, F. W., & Saber, E. (2016). Classification of remote sensed images using random forests and deep learning framework. *SPIE,* .

Pire, T., Fischer, T., Civera, J., de Cristóforis, P., & Jacobo-Berlles, J. (2015). Stereo parallel tracking and mapping for robot localization. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS).*

Pirsiavash, H., Ramanan, D., & Fowlkes, C. C. (2011). Globally-optimal greedy algorithms for tracking a variable number of objects. In *Proc. IEEE*

Conf. on Computer Vision and Pattern Recognition (CVPR).

Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P. V., & Schiele, B. (2016). Deepcut: Joint subset partition and labeling for multi person pose estimation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Pishchulin, L., Jain, A., Andriluka, M., Thormaehlen, T., & Schiele, B. (2012). Articulated people detection and pose estimation: Reshaping the future. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Plotkin, L. (2015). *PyDriver: Entwicklung eines Frameworks für räumliche Detektion und Klassifikation von Objekten in Fahrzeugumgebung.* Master's thesis Karlsruhe Institute of Technology.

Plyer, A., Besnerais, G. L., & Champagnat, F. (2014). Massively parallel lucas kanade optical flow for real-time video processing applications. *Journal of Real-Time Image Processing (JRTIP),* (pp. 1–18).

Pohlen, T., Hermans, A., Mathias, M., & Leibe, B. (2016). Full-resolution residual networks for semantic segmentation in street scenes. *arXiv.org, 1611.08323.*

Pollard, T., & Mundy, J. L. (2007). Change detection in a 3-d world. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Pollefeys, M. (2008). Detailed real-time urban 3d reconstruction from video. *International Journal of Computer Vision (IJCV), 78,* 143–167.

Pomerleau, D., & Jochem, T. (1996). Rapidly adapting machine vision for automated vehicle steering. *EXPERT,* .

Premebida, C., Carreira, J., Batista, J., & Nunes, U. (2014). Pedestrian detection combining rgb and dense lidar data. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS).*

Qiu, W., & Yuille, A. L. (2016). Unrealcv: Connecting computer vision to unreal engine. *arXiv.org, 1609.01326.*

Quam, L. H. (1984). Hierarchical warp stereo. In *Image Understanding Workshop (IUW).*

Quang, N. T., Thuy, N. T., Sang, D. V., & Binh, H. T. T. (2015). *Semantic Segmentation for Aerial Images using RF and a full-CRF.* Technical Report Ha Noi University of Science and Technology, Vietnam.

Rabe, C., Mueller, T., Wedel, A., & Franke, U. (2010). Dense, robust, and accurate motion field estimation from stereo image sequences in real-time. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Ranftl, R., Bredies, K., & Pock, T. (2014). Non-local total generalized variation for optical flow estimation. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Ranftl, R., Pock, T., & Bischof, H. (2013). Minimizing TGV-based variational models with non-convex data terms. In *Proc. of the International Conf. on Scale Space and Variational Methods in Computer Vision (SSVM).*

Ranjan, A., & Black, M. J. (2016). Optical flow estimation using a spatial pyramid network. *arXiv.org, 1611.00850.*

Rebecq, H., Horstschaefer, T., Gallego, G., & Scaramuzza, D. (2016). Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real-time. In *IEEE Robotics and Automation Letters (RA-L).*

Ren, S., He, K., Girshick, R. B., & Sun, J. (2015). Faster R-CNN: towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS).*

Ren, X., & Malik, J. (2003). Learning a classification model for segmentation. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV).*

Revaud, J., Weinzaepfel, P., Harchaoui, Z., & Schmid, C. (2015). EpicFlow: Edge-preserving interpolation of correspondences for optical flow. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Rezatofighi, S. H., Milan, A., Zhang, Z., Shi, Q., Dick, A., & Reid, I. (2015). Joint probabilistic data association revisited. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV).*

Rhemann, C., Hosni, A., Bleyer, M., Rother, C., & Gelautz, M. (2011). Fast cost-volume filtering for visual correspondence and beyond. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Richardt, C., Kim, H., Valgaerts, L., & Theobalt, C. (2016). Dense wide-baseline scene flow from two handheld video cameras. In *Proc. of the International Conf. on 3D Vision (3DV).*

Richter, S. R., Vineet, V., Roth, S., & Koltun, V. (2016). Playing for data: Ground truth from computer games. In *Proc. of the European Conf. on Computer Vision (ECCV).*

Riegler, G., Ulusoy, A. O., & Geiger, A. (2017). Octnet: Learning deep 3d representations at high resolutions. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

Riemenschneider, H., Bódis-Szomorú, A., Weissenberg, J., & Gool, L. V. (2014). Learning where to classify in multi-view semantic segmentation. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*.

Ros, G., Sellart, L., Materzynska, J., Vazquez, D., & Lopez, A. (2016). The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Rottensteiner, F., Sohn, G., Gerke, M., & Wegner, J. D. (2013). *ISPRS Test Project on Urban Classification and 3D Building Reconstruction*. Technical Report ISPRS Commission III.

Rottensteiner, F., Sohn, G., Gerke, M., Wegner, J. D., Breitkopf, U., & Jung, J. (2014). Results of the {ISPRS} benchmark on urban object detection and 3d building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing (JPRS)*, *93*, 256 – 271.

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). Orb: an efficient alternative to sift or surf. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)* (pp. 2564–2571).

Sanchez-Matilla, R., Poiesi, F., & Cavallaro, A. (2016). Online multi-target tracking with strong and weak detections. In *Proc. of the European Conf. on Computer Vision (ECCV) Workshops*.

Sattler, T., Havlena, M., Radenovic, F., Schindler, K., & Pollefeys, M. (2015). Hyperpoints and fine vocabularies for large-scale location recognition. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Sattler, T., Leibe, B., & Kobbelt, L. (2016). Efficient effective prioritized matching for large-scale image-based localization. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *PP*, 1–1.

Savva, M., Chang, A. X., & Hanrahan, P. (2015). Semantically-enriched 3d models for common-sense knowledge. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) Workshops*, .

Scaramuzza, D., & Fraundorfer, F. (2011). Visual odometry [tutorial]. *Robotics and Automation Magazine (RAM)*, *18*, 80–92.

Scaramuzza, D., Fraundorfer, F., & Siegwart, R. (2009). Real-time monocular visual odometry for on-road vehicles with 1-point RANSAC. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Scaramuzza, D., & Martinelli, A. (2006). A toolbox for easily calibrating omnidirectional cameras. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*.

Scaramuzza, D., & Siegwart, R. (2008). Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles. *IEEE Trans. on Robotics*, *24*, 1015–1026.

Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nesic, N., Wang, X., & Westling, P. (2014). High-resolution stereo datasets with subpixel-accurate ground truth. In *Proc. of the German Conference on Pattern Recognition (GCPR)*.

Scharstein, D., & Szeliski, R. (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision (IJCV)*, *47*, 7–42.

Scharstein, D., & Szeliski, R. (2003). High-accuracy stereo depth maps using structured light. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Schneider, L., Cordts, M., Rehfeld, T., Pfeiffer, D., Enzweiler, M., Franke, U., Pollefeys, M., & Roth, S. (2016). Semantic stixels: Depth is not enough. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Schreiber, M., Knöppel, C., & Franke, U. (2013). Laneloc: Lane marking based localization using highly accurate maps. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Schönbein, M., & Geiger, A. (2014). Omnidirectional 3d reconstruction in augmented manhattan worlds. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*.

Schönbein, M., Strauss, T., & Geiger, A. (2014). Calibrating and centering quasi-central catadioptric cameras. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Schöps, T., Schönberger, J., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., & Geiger, A. (2017). A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Seff, A., & Xiao, J. (2016). Learning from maps: Visual common sense for autonomous driving. *arXiv.org*, *1611.08583*.

Seitz, S. M., Curless, B., Diebel, J., Scharstein, D., & Szeliski, R. (2006). A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Seki, A., & Pollefeys, M. (2016). Patch based confidence prediction for dense disparity map. In *Proc. of the British Machine Vision Conf. (BMVC)*.

Sengupta, S., Greveson, E., Shahrokni, A., & Torr, P. H. (2013). Urban 3d semantic modelling using stereo vision. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Sengupta, S., Sturgess, P., Ladicky, L., & Torr, P. H. S. (2012). Automatic dense visual semantic mapping from street-level imagery. In *IROS*.

Sermanet, P., Kavukcuoglu, K., Chintala, S., & LeCun, Y. (2013). Pedestrian detection with unsupervised multi-stage feature learning. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Sevilla-Lara, L., Sun, D., Jampani, V., & Black, M. J. (2016). Optical flow with semantic segmentation and localized layers. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Shan, Q., Wu, C., Curless, B., Furukawa, Y., Hernández, C., & Seitz, S. M. (2014). Accurate geo-registration by ground-to-aerial image matching. In *Proc. of the International Conf. on 3D Vision (3DV)*.

Sharifzadeh, S., Chiotellis, I., Triebel, R., & Cremers, D. (2016). Learning to drive using inverse reinforcement learning and deep q-networks. In *Advances in Neural Information Processing Systems (NIPS) Workshops*.

Shashua, A., Gdalyahu, Y., & Hayun, G. (2004). Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*.

Sherrah, J. (2016). Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery. *arXiv.org*, *1606.02585*.

Shotton, J., Winn, J., Rother, C., & Criminisi, A. (2009). Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *International Journal of Computer Vision (IJCV)*, *81*, 2–23.

Simoncelli, E. P., Adelson, E. H., & Heeger, D. J. (1991). Probability distributions of optical flow. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *Proc. of the International Conf. on Learning Representations (ICLR)*.

Smith, R., Self, M., & Cheeseman, P. (1987). Estimating uncertain spatial relationships in robotics. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Song, S., & Chandraker, M. (2014). Robust scale estimation in real-time monocular SFM for autonomous driving. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Song, S., Chandraker, M., & Guest, C. C. (2013). Parallel, real-time monocular visual odometry. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

m. Song, Y., & Jeon, M. (2016). Online multiple object tracking with the hierarchically adopted gm-phd filter using motion and appearance. In *Proc. of International Conf. on Consumer Electronics-Asia (ICCE Asia)*.

Speldekamp, T., Fries, C., Gevaert, C., & Gerke, M. (2015). *Automatic Semantic Labelling of Urban Areas using a rule-based approach and realized with MeVisLab*. Technical Report University of Twente.

Stiefelhagen, R., Bernardin, K., Bowers, R., Garofolo, J., Mostefa, D., & Soundararajan, P. (2007). The clear 2006 evaluation. In *CLEAR*.

Strasdat, H., Davison, A. J., Montiel, J. M. M., & Konolige, K. (2011). Double window optimisation for constant time visual SLAM. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)* (pp. 2352–2359).

Strasdat, H., Montiel, J. M. M., & Davison, A. J. (2010). Scale drift-aware large scale monocular SLAM. In *Proc. Robotics: Science and Systems (RSS)*.

Suleymanov, T., Paz, L. M., Piniés, P., Hester, G., & Newman, P. (2016). The path less taken: A fast variational approach for scene segmentation used for closed loop control. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*.

Sun, D., Roth, S., & Black, M. J. (2014). A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision (IJCV)*, *106*, 115–137.

Sun, M., & Savarese, S. (2011). Articulated part-based model for joint object detection and pose estimation. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. E., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions.

In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Tang, S., Andres, B., Andriluka, M., & Schiele, B. (2016a). Multi-person tracking by multicut and deep matching. In *Proc. of the European Conf. on Computer Vision (ECCV) Workshops*.

Tang, S., Andres, B., Andriluka, M., & Schiele, B. (2016b). Multi-person tracking by multicut and deep matching. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Tardif, J., George, M. D., Laverne, M., Kelly, A., & Stentz, A. (2010). A new approach to vision-aided inertial navigation. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*.

Thorpe, C., Hebert, M. H., Kanade, T., & Shafer, S. A. (1988). Vision and navigation for the carnegie-mellon navlab. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *10*, 362–372.

Timofte, R., & Gool, L. V. (2015). Sparse flow: Sparse matching for small to large displacement optical flow. In *Proc. of the IEEE Winter Conference on Applications of Computer Vision (WACV)*.

Tschannen, M., Cavigelli, L., Mentzer, F., Wiatowski, T., & Benini, L. (2016). Deep structured features for semantic segmentation. *arXiv.org*, *1609.07916*.

Tu, Z., & Bai, X. (2010). Auto-context and its application to high-level vision tasks and 3d brain image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *32*, 1744–1757.

Uhrig, J., Cordts, M., Franke, U., & Brox, T. (2016). Pixel-level encoding and depth layering for instance-level semantic labeling. In *Proc. of the German Conference on Pattern Recognition (GCPR)* (pp. 14–25).

Uijlings, J. R., van de Sande, K. E., Gevers, T., & Smeulders, A. W. (2013). Selective search for object recognition. *International Journal of Computer Vision (IJCV)*, *104*, 154–171.

Ulusoy, A. O., Geiger, A., & Black, M. J. (2015). Towards probabilistic volumetric reconstruction using ray potentials. In *Proc. of the International Conf. on 3D Vision (3DV)*.

Uras, S., Girosi, F., Verri, A., & Torre, V. (1988). A computational approach to motion perception. *Biological Cybernetics*, *60*, 79–87.

Valentin, J. P., Sengupta, S., Warrell, J., Shahrokni, A., & Torr, P. H. (2013). Mesh based semantic modelling for indoor and outdoor scenes. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Vedula, S., Baker, S., Rander, P., Collins, R., & Kanade, T. (1999). Three-dimensional scene flow. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Verdie, Y., & Lafarge, F. (2014). Detecting parametric objects in large scenes by monte carlo sampling. *International Journal of Computer Vision (IJCV)*, *106*, 57–75.

Vijayanarasimhan, S., & Grauman, K. (2012). Active frame selection for label propagation in videos. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Vineet, V., Miksik, O., Lidegaard, M., Niessner, M., Golodetz, S., Prisacariu, V. A., Kahler, O., Murray, D. W., Izadi, S., Perez, P., & Torr, P. H. S. (2015). Incremental dense semantic stereo fusion for large-scale semantic scene reconstruction. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Viola, P. A., & Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision (IJCV)*, *57*, 137–154.

Viola, P. A., Jones, M. J., & Snow, D. (2005). Detecting pedestrians using patterns of motion and appearance. *International Journal of Computer Vision (IJCV)*, *63(2)*, 153–161.

Vogel, C., Roth, S., & Schindler, K. (2013). An evaluation of data costs for optical flow. In *Proc. of the German Conference on Pattern Recognition (GCPR)*.

Vogel, C., Schindler, K., & Roth, S. (2015). 3d scene flow estimation with a piecewise rigid scene model. *International Journal of Computer Vision (IJCV)*, *115*, 1–28.

Volpi, M., & Tuia, D. (2016). Dense semantic labeling of sub-decimeter resolution images with convolutional neural networks. *arXiv.org*, *abs/1608.00775*.

Walch, F., Hazirbas, C., Leal-Taixé, L., Sattler, T., Hilsenbeck, S., & Cremers, D. (2016). Image-based localization with spatial lstms. *arXiv.org*, *1611.07890*.

Wang, D. Z., & Posner, I. (2015). Voting for voting in online point cloud object detection. In *Proc. Robotics: Science and Systems (RSS)*.

Wang, S., Bai, M., Mattyus, G., Chu, H., Luo, W., Yang, B., Liang, J., Cheverie, J., Fidler, S., & Urtasun, R. (2016). Torontocity: Seeing the world with a million eyes. *arXiv.org*, *1612.00423*.

Wang, X., Yang, M., Zhu, S., & Lin, Y. (2015). Regionlets for generic object detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *37*, 2071–2084.

Wedel, A., Brox, T., Vaudrey, T., Rabe, C., Franke, U., & Cremers, D. (2011). Stereoscopic scene flow computation for 3D motion understanding. *International Journal of Computer Vision (IJCV)*, *95*, 29–51.

Wedel, A., Rabe, C., Badino, H., Loose, H., Franke, U., & Cremers, D. (2009). B-spline modeling of road surfaces with an application to free space estimation. *IEEE Trans. on Intelligent Transportation Systems (TITS)*, *10*, 572–583.

Wedel, A., Rabe, C., Vaudrey, T., Brox, T., Franke, U., & Cremers, D. (2008). Efficient dense scene flow from sparse or dense stereo data. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Wegner, J. D., Branson, S., Hall, D., Schindler, K., & Perona, P. (2016). Cataloging public objects using aerial and street-level images - urban trees. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Wegner, J. D., Montoya-Zegarra, J. A., & Schindler, K. (2013). A higher-order CRF model for road network extraction. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Wegner, J. D., Montoya-Zegarra, J. A., & Schindler, K. (2015). Road networks as collections of minimum cost paths. *ISPRS Journal of Photogrammetry and Remote Sensing (JPRS)*, *108*, 128–137.

Wei, D., Liu, C., & Freeman, W. (2014). A data-driven regularization model for stereo and flow. In *Proc. of the International Conf. on 3D Vision (3DV)*.

Weinzaepfel, P., Revaud, J., Harchaoui, Z., & Schmid, C. (2013). DeepFlow: Large displacement optical flow with deep matching. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Whelan, T., Leutenegger, S., Salas-Moreno, R. F., Glocker, B., & Davison, A. J. (2015). Elasticfusion: Dense SLAM without A pose graph. In *Proc. Robotics: Science and Systems (RSS)*.

Winner, H., Hakuli, S., Lotz, F., Singer, C., Geiger, A. et al. (2015). *Handbook of Driver Assistance Systems*. Springer Vieweg.

Wojek, C., Roth, S., Schindler, K., & Schiele, B. (2010). Monocular 3d scene modeling and inference: Understanding multi-object traffic scenes. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Wojek, C., & Schiele, B. (2008a). A dynamic conditional random field model for joint labeling of object and scene classes. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Wojek, C., & Schiele, B. (2008b). A performance evaluation of single and multi-feature people detection. In *Proc. of the DAGM Symposium on Pattern Recognition (DAGM)*.

Wojek, C., Walk, S., Roth, S., & Schiele, B. (2011). Monocular 3d scene understanding with explicit occlusion reasoning. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Wojek, C., Walk, S., Roth, S., Schindler, K., & Schiele, B. (2013). Monocular visual scene understanding: Understanding multi-object traffic scenes. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *35*, 882–897.

Wojek, C., Walk, S., & Schiele, B. (2009). Multi-cue onboard pedestrian detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Wolff, M., Collins, R. T., & Liu, Y. (2016). Regularity-driven facade matching between aerial and street views. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Workman, S., Souvenir, R., & Jacobs, N. (2015). Wide-area image geolocalization with aerial reference imagery. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Wu, T., Li, B., & Zhu, S. (2016a). Learning and-or model to represent context and occlusion for car detection and viewpoint estimation. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *38*, 1829–1843.

Wu, Z., Shen, C., & van den Hengel, A. (2016b). Wider or deeper: Revisiting the resnet model for visual recognition. *arXiv.org*, *1611.10080*.

Wulff, J., & Black, M. J. (2015). Efficient sparse-to-dense optical flow estimation using a learned basis and layers. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Xiang, Y., Alahi, A., & Savarese, S. (2015a). Learning to track: Online multi-object tracking by decision making. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Xiang, Y., Choi, W., Lin, Y., & Savarese, S. (2015b). Data-driven 3d voxel patterns for object category recognition. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Xiang, Y., Choi, W., Lin, Y., & Savarese, S. (2016). Subcategory-aware convolutional neural networks for object proposals and detection. *arXiv.org*, *1604.04693*.

Xiao, J., Fang, T., Zhao, P., Lhuillier, M., & Quan, L. (2009). Image-based street-side city modeling. *ACM Trans. on Graphics (SIGGRAPH)*, *28*, 114:1–114:12.

Xiao, J., & Quan, L. (2009). Multiple view semantic segmentation for street view images. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Xie, J., Kiefel, M., Sun, M.-T., & Geiger, A. (2016). Semantic instance annotation of street scenes by 3d to 2d label transfer. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Xu, H., Gao, Y., Yu, F., & Darrel, T. (2016). End-to-end learning of driving models from large-scale video dataset. *arXiv.org*, *1612.01079*.

Yamaguchi, K., Hazan, T., McAllester, D., & Urtasun, R. (2012). Continuous markov random fields for robust stereo estimation. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Yamaguchi, K., McAllester, D., & Urtasun, R. (2013). Robust monocular epipolar flow estimation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Yamaguchi, K., McAllester, D., & Urtasun, R. (2014). Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Yang, B., & Nevatia, R. (2012). An online learned crf model for multi-target tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Yang, F., Choi, W., & Lin, Y. (2016). Exploit all the layers: Fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Yang, Y., Baker, S., Kannan, A., & Ramanan, D. (2012). Recognizing proxemics in personal photos. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (pp. 3522–3529).

Yoon, J. H., Lee, C.-R., Yang, M.-H., & Yoon, K.-J. (2016). Online multi-object tracking via structural constraint event aggregation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Yoon, J. H., Yang, M., Lim, J., & Yoon, K. (2015). Bayesian multi-object tracking using motion context from multiple objects. In *Proc. of the IEEE Winter Conference on Applications of Computer Vision (WACV)*.

Yu, F., & Koltun, V. (2016). Multi-scale context aggregation by dilated convolutions. In *Proc. of the International Conf. on Learning Representations (ICLR)*.

Yu, F., Li, W., Li, Q., Liu, Y., Shi, X., & Yan, J. (2016). Poi: Multiple object tracking with high performance detection and appearance feature. In *Proc. of the European Conf. on Computer Vision (ECCV) Workshops*.

Yu, F., Xiao, J., & Funkhouser, T. A. (2015). Semantic alignment of lidar data at city scale. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (pp. 1722–1731).

Zach, C., Pock, T., & Bischof, H. (2007). A duality based approach for realtime TV-L1 optical flow. In *Proc. of the DAGM Symposium on Pattern Recognition (DAGM)* (pp. 214–223).

Žbontar, J., & LeCun, Y. (2016). Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research (JMLR)*, *17*, 1–32.

Zhang, H., Geiger, A., & Urtasun, R. (2013). Understanding high-level semantics by modeling traffic patterns. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Zhang, J., Kaess, M., & Singh, S. (2014). Real-time depth enhanced monocular odometry. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*.

Zhang, J., & Singh, S. (2014). LOAM: lidar odometry and mapping in real-time. In *Proc. Robotics: Science and Systems (RSS)*.

Zhang, J., & Singh, S. (2015). Visual-lidar odometry and mapping: low-drift, robust, and fast. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*.

Zhang, L., Li, Y., & Nevatia, R. (2008). Global data association for multi-object tracking using network flows. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, L., Lin, L., Liang, X., & He, K. (2016a). Is faster r-cnn doing well for pedestrian detection? In *Proc. of the European Conf. on Computer Vision (ECCV)*.

Zhang, S., Benenson, R., Omran, M., Hosang, J. H., & Schiele, B. (2016b). How far are we from solving pedestrian detection? In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, Z., Fidler, S., & Urtasun, R. (2016c). Instance-level segmentation for autonomous driving with deep densely connected mrfs. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, Z., Schwing, A. G., Fidler, S., & Urtasun, R. (2015). Monocular object instance segmentation and depth ordering with cnns. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2016). Pyramid scene parsing network. *arXiv.org*, *1612.01105*.

Zheng, Y., Zhao, M., Song, Y., Adam, H., Buddemeier, U., Bissacco, A., Brucher, F., Chua, T.-S., & Neven, H. (2009). Tour the world: Building a web-scale landmark recognition engine. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., & Oliva, A. (2014). Learning Deep Features for Scene Recognition using Places Database. In *Advances in Neural Information Processing Systems (NIPS)*.

Zhou, C., Güney, F., Wang, Y., & Geiger, A. (2015). Exploiting object similarity in 3d reconstruction. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*.

Zhu, H., Yuen, K. V., Mihaylova, L., & Leung, H. (2017). Overview of environment perception for intelligent vehicles. *IEEE Trans. on Intelligent Transportation Systems (TITS)*, *PP*, 1–18.

Zhu, Y., Wang, J., Zhao, C., Guo, H., & Lu, H. (2016). Scale-adaptive deconvolutional regression network for pedestrian detection. In *Proc. of the Asian Conf. on Computer Vision (ACCV)*.

Zia, M., Stark, M., Schiele, B., & Schindler, K. (2013). Detailed 3D representations for object recognition and modeling. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, *35*, 2608–2623.

Zia, M., Stark, M., & Schindler, K. (2015). Towards scene understanding with detailed 3d object representations. *International Journal of Computer Vision (IJCV)*, *112*, 188–203.

Ziegler, J., Bender, P., Schreiber, M., & Lategahnf, H. (2014). Making bertha drive - an autonomous journey on a historic route. *Proc. IEEE Intelligent Transportation Systems Magazine (ITSM)*, *6*, 8–20.

Zimmer, H., Bruhn, A., & Weickert, J. (2011). Optic flow in harmony. *International Journal of Computer Vision (IJCV)*, *93*, 368–388.