Key Technologies for Intelligent and Safer Cars – from Motion Estimation to Predictive Collision Avoidance

Davide Scaramuzza, Luciano Spinello, Rudolph Triebel, Roland Siegwart

Autonomous Systems Lab
ETH Zurich

Abstract: One of the research areas that has received more and more interest during the last years is the development of driver assistant systems and semi-autonomous cars. However, densely populated environments like city centers are still a challenge for the operation of such systems. In this paper, we present approaches to two of the major tasks for autonomous driving in urban environments: self-localization and ego-motion estimation and detection of dynamic objects such as cars and pedestrians. For each of these tasks we present a summary of the techniques we employ and results on real data. All modules have been implemented and tested on our autonomous car platform SmartTer.

I. Introduction

Over the last two decades, we have assisted to a rapid research progress in driver assistance systems. Some of these systems have even reached the market and have become nowadays an essential tool for driving. GPS navigation systems are probably the most significant ones. They have revolutionized the way of traveling and certainly facilitated research towards fully autonomous navigation in outdoor environments. Results of autonomous driving in a mock urban environment have been very successfully demonstrated during the 2007 DARPA Urban Challenge [1] where vehicles had to navigate autonomously while obeying all traffic regulations. Additionally, they had to take "intelligent" decisions in real-time based on the actions of other vehicles like for negotiating priorities. However, there are still numerous challenges that have to be solved in view of fully autonomous navigation of cars in very cluttered environments. Especially in city centers, where many different kinds of transportation systems are encountered (walking, cycling, driving, etc.), the requirements for an autonomous system are very high. The key prerequisites for such systems are localization and ego-motion estimation and reliable detection and tracking of dynamic objects.

Both localization and dynamic obstacle detection are challenging. Even when GPS is available, localization accuracy can become as bad as 50 meters in urban areas. This prevents the vehicle from accurately recognizing where it is. Additionally, detection of pedestrians and other vehicles is still nowadays far from being failure-free. Compared to vehicles, pedestrians are obviously very vulnerable as they are not endowed with protections. According to the annual traffic accident statistics published by the Touring Club Switzerland (TCS) [1], over the last 30 years there has been a decrease in the number of dead and seriously injured persons due to the growing availability of driving safety systems. At the same time, however, we have also seen an increase in the number of dead and injured pedestrians due to the fact that it is usually motorists and cyclists, but not pedestrians, who benefit from such safety systems. One way to tackle this problem is therefore to build more intelligent cars able to avoid potential collision with pedestrians.

In this paper, we address the problem of localization and ego-motion estimation and reliable detection and tracking of dynamic objects. For each of these tasks we present new approaches. They have been implemented and tested on an autonomous robotic platform based on a Smart car that is equipped with several sensors (see Fig. 1, left).

This paper is organized as follows. In Sec. II we present our approach to estimate the ego-motion of the vehicle by just using the visual input from a single omnidirectional camera. Sec. III describes our algorithm to detect cars and pedestrians from camera and 2D laser data. Finally, Sec. IV concludes the paper.

II. Monocular Visual Motion Estimation

A. Overview

In this section, we will show how a single omnidirectional camera can be used for accurate motion estimation and mapping without using the information from any other sensors.

The problem of recovering relative camera poses and 3D structure from a set of monocular images has been

[1] http://www.tcs.ch/main/de/home/sicherheit/infrastrukturen/statistik unfalle.html

Fig. 1. The autonomous robot *SmartTer* developed at the Autonomous Systems Lab at the ETH Zurich. For the ego-motion estimation presented in this paper, we use the omnidirectional camera mounted on the roof, while the front laser, the perspective camera, and the rotating 3D scanner are used for detection and tracking of pedestrians and cars. The other static lasers are for collision avoidance, which is not covered in this paper.

largely studied for many years and is known in the computer vision community as "Structure From Motion" (SFM) [2] or *visual odometry*. Successful results with only a single camera and over long distances (from hundreds of meters up to kilomeers) have been obtained in the last decade using both perspective and omnidirectional cameras (see [3], [4], [5], [6], [7], [8], [9], [10], [11], [12]).

Closely related to structure from motion is what is known in the robotics community as Simultaneous Localization and Mapping (SLAM), which aims at estimating the motion of the robot while simultaneously building and updating the environment map. SLAM has been most often performed with other sensors than regular cameras, however in the last years successful results have been obtained using single cameras alone (see [13], [14], [15], [16]).

The term *visual odometry* was coined only in 2004 by Nister [6] who presented the first visual odometry system using a binocular camera (i.e. stereo camera). In this section, conversely, we are concerned about monocular camera. This problem involves extracting point correspondences between two camera images and using well known theory from Algebra and Geometry to determine both camera displacement and 3D structure up to a scale. The absolute scale can be determined by using multiple cameras or using the information from other sensors. The minimum number of points to estimate motion and structure is 5, as 5 is the number of parameters which describe the unconstrained motion up to a scale (in fact, we have 6 degrees of freedom minus the unknown scale).

Using cameras instead of other sensors for computing ego-motion allows for a simple integration of ego-motion data into other vision based algorithms, such as obstacle, pedestrian, and car detection, without the need for calibration between sensors. This reduces maintenance and cost. Furthermore, vision has been shown to provide better motion estimates than wheel encoders. Because of this, automobile industries are considering to integrate visual odometry systems in future generation cars. This will be used to replace GPS in GPS denied environments or when the GPS information is not reliable due to multipath or poor satellite coverage. Other vehicle on-board sensors like compass and IMU will be used to boost the estimation in case of poor visibility or unfavorable conditions.

While there exist nowadays a wide availability of algorithms for motion estimation using video input alone, cameras are still little integrated in the motion estimation system of a mobile robot and even less in that of an automotive vehicle. The main reasons for this are the following:

- several algorithms can still only work off-line or at low frame-rate,
- others need high processing power or expensive and dedicated processors,
- many algorithms are quite complex to use or are designed for specific cameras,
- many algorithms assume static scenes and cannot cope with dynamic and cluttered environments or huge occlusions by other passing vehicles (like what happens in typical urban environments in real traffic with other moving cars, buses, trams and pedestrians, sudden changes of speed, etc.),
- the data-association problem (feature matching and outlier removal) is not completely robust and can fail,
- the motion estimation scheme usually requires many keypoints and can fail when only a few keypoints are available in almost absence of structure.

To recap, visual odometry is a *data association* problem. As shown in the literature, the best results are obtained with an omnidirectional camera as features can be tracked longer and more robustly over time. However, the biggest problem of visual odometry remains and is *data association*. Indeed, matched points contain many outliers that must be detected and removed for the motion to be accurately estimated. In the last few years, a very established method for removing outliers has been the "5-point RANSAC" algorithm, developed by Nister [17], which needs a minimum of 5 point correspondences to estimate the model hypotheses. Because of this, however, it can require from several hundreds up to thousands of iterations

to find a set of points free of outliers. In the best implementation, this algorithm runs between 10-20 Hz, which is still too slow for automotive applications.

In our previous work [10], [11], [12], we showed that all the above mentioned areas can be improved by using a restrictive motion model which allows us to parameterize the motion with only 1 feature correspondence. Using a single feature correspondence for motion estimation is the lowest model parameterization possible and results in the most efficient algorithms for outlier removal and motion estimation.

B. Our Approach: Exploiting Nonholonomic Constraints

Our approach exploits the nonholonomic constraints of wheeled vehicles, that is, they possess an Instantaneous Center of Rotation (ICR). Cars are typical examples of such vehicles. As everybody experiences in driving, one needs to act on the steering to change the direction of the car. What actually happens in practice is that the two front wheels are turned of a slight different angle to make the vehicle move instantaneously along a circle and, thus, turn about the ICR (Fig. 2). As the reader can perceive, the motion of a camera installed on the vehicle can be then locally described with circular motion; straight motion can be represented along a circle of infinite radius. This constraint reduces the degrees of freedom of the motion to two, namely the rotation angle and the radius of curvature. The first consequence is that only one feature correspondence suffices for computing the epipolar geometry (in fact up-to-scale circular motion is described only by the rotation angle). This allows motion to be computed also from scenes where structure is almost absent, provided that at least one feature is available. The second consequence is a very efficient method for removing outliers can be implemented, which we called "1-Point RANSAC" [11]. Using our motion estimation algorithm we were able to process frames up 400 Hz, which is, to the best of our knowledge, the most efficient motion estimation algorithm.

A detailed description of our 1-Point RANSAC algorithm as well as the mathematical derivation of the motion parameters using the nonholonomic constraints can be found in the our previous work [11] and its follow-up [12]. In the latter, we show that by exploiting the nonholonomic constraints we can even estimate the absolute scale from a single camera without using any user input, nor the odometry of the vehicle.

C. Results

Our motion estimation method has been successfully tested on our autonomous car (a Smart) which is equipped with an omnidirectional camera. A picture of our vehicle with the omnidirectional camera is shown in Fig. 1. Our omnidirectional camera is composed of

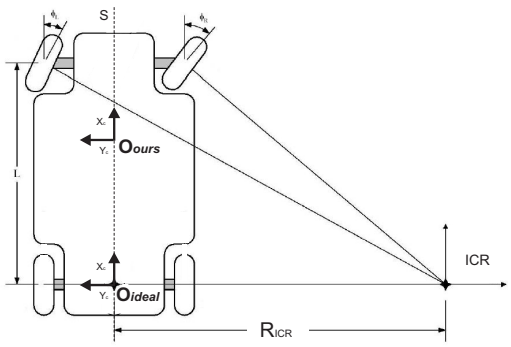

Fig. 2. General Ackermann steering principle.

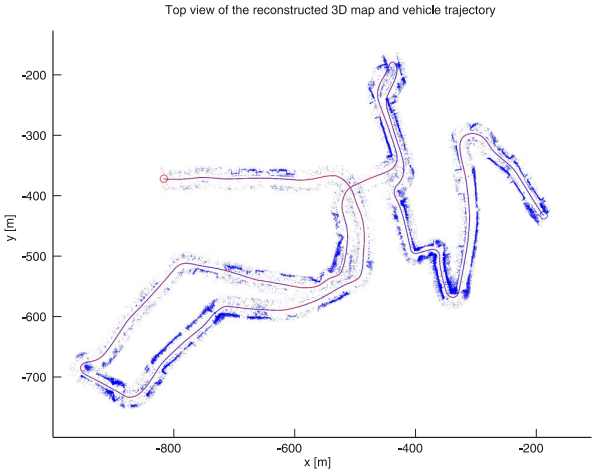

Fig. 3. Recovered 3D map and camera positions: top view.

a hyperbolic mirror (KAIDAN 360 One VR) and a digital color camera (SONY XCD-SX910, image size 640×480 pixels). For calibrating the camera we used the toolbox described in [18] and available from [19]. The vehicle speed ranged between 0 and 45Km/h.

The dataset was taken in real traffic during the peak time in the city center of Zurich. Therefore, many pedestrians and passing trams, buses, and cars were also present. The images were collected from the beginning until the end of the tour, also when the vehicle was still in the presence of stop signs, pedestrian crossings, and red lights. The video sequence used in the experiments as well as the final motion estimation result can be watched on the first author's website. The overall length of the tour was about 3Km and is shown in Fig. 4 overlaid on a satellite image.

Motion estimation was done by triangulating feature points, tracking them, and estimating new poses from them up to a scale. The absolute scale was computed using our method [12].

We tested our structure from motion algorithm on different feature detectors: SIFT, Harris, KLT, and FAST corners. SIFT returned about $700 \sim 1000$ features per frame, while Harris, KLT, FAST about

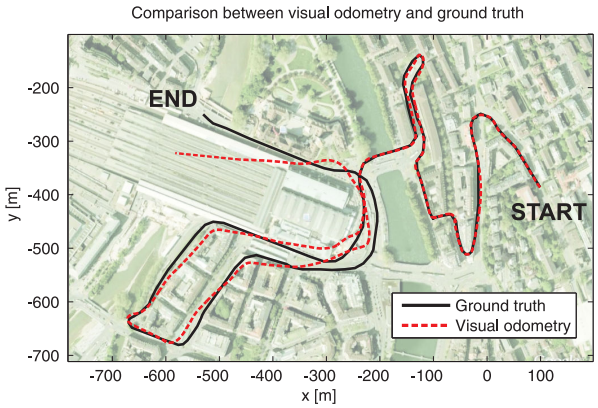

Fig. 4. Comparison between visual odometry (red dashed line) and ground truth (black solid line). The entire trajectory is 3Km long.

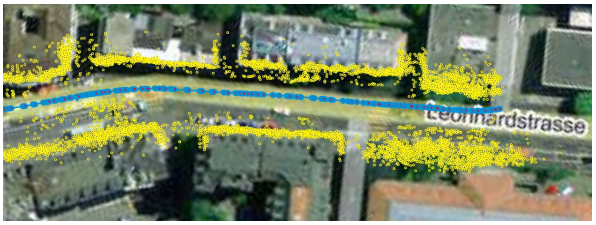

Fig. 5. A close-up of the recovered 3D map (yellow) overlaid on a satellite image. Camera positions are in blue. This image represents the street Leonhardstrasse in Zurich. The cluttered points at the beginning of the path on the right are trees.

$2500 \sim 4000$ features.

Figure 3 shows the top view of the recovered 3D map and camera trajectory. Furthermore observe that the points are aligned quite well along straight edges, which correspond to the walls of the buildings. Finally, Fig. 5 shows a closer view of the 3D map at the beginning of the path overlaid on a satellite image. Here it is more clear that the 3D points are well aligned along the straight edges of the buildings.

The best results, in terms of agreement with the ground truth, were obtained with Harris, KLT, or FAST features, mainly because of their high density. The comparison is shown in Fig. 4. As observed the path is aligned quite well with the real trajectory except for the unavoidable drift that increases with the traveled distance. This result is however very good if one considers that the proposed approach is incremental (the motion was estimated only between two consecutive views without skipping frames and without correcting the previous poses) and that the position of the triangulated features is unchanged. Furthermore, notice that we did not apply bundle-adustment. Bundle-adjustments would correct both camera poses and feature positions and the final map would appear greatly improved. Furthermore, consider that the length of the recovered path was considerably long, 3Km. Loop detection and SLAM could be used to remove the motion drift. These improvements are currently under development. Some results on loop closing using vocabulary trees are reported in [20].

III. Pedestrian and Car Detection

A. Overview

The system we employ to detect pedestrians and cars consists of three main components: an appearance based detector that uses the information from camera images, a 2D-laser based detector providing structural information, and a tracking module that uses the combined information from both sensors and estimates the motion vector for each tracked object. The laser based detection applies a boosted Conditional Random Field (CRF) on geometrical and statistical features of 2D scan points. The image based detector uses and extended version of the Implicit Shape Model (ISM) [21]. It operates on a region of interest obtained from projecting the laser detection into the image to constrain the position and scale of the objects. The tracking module applies an Extended Kalman Filter (EKF) with two motion models, fusing the information from camera and laser. A detailed description of the overall system can be found in [22], here we only give a short summary.

B. Our Approach: Use of Machine Learning Techniques

Our appearance-based people detector is based on scale-invariant Implicit Shape Models (ISM) [21]. In summary, an ISM consists in a set of local region descriptors, called the *codebook*, and a set of displacements and scale factors, usually named *votes*, for each descriptor. A vote points from the position of the descriptor to the center of the object as it was found in the training data. The codebook with the votes is first learned from a labeled training data set and then used for detection. In the past, we presented several improvements to the standard ISM approach [22], [23], [24], where the most recent ones are *sub-part detection*, extraction of *template masks* and the defintion of *superfeatures* (see Fig. 6). The main idea behind all these extensions is to enrich and to refine the information extracted from the training data, leading to voters that can distinguish between object sub-parts, in- and outside of object masks, as well as weak and strong features.

The laser-based detector uses Conditional Random Fields (CRFs) [25], which represent the conditional probability $p(\mathbf{y} \mid \mathbf{x})$ using an undirected cyclic graph, where each node is associated with a label y_i (pedestrian, car, background) and a feature vector $\mathbf{x}_i$. The edges model the conditional dependency of two neighboring data points and have also associated feature vectors. The node features include size, circularity, standard deviation, etc. We apply AdaBoost [26] to these feature vectors to account for the non-linear

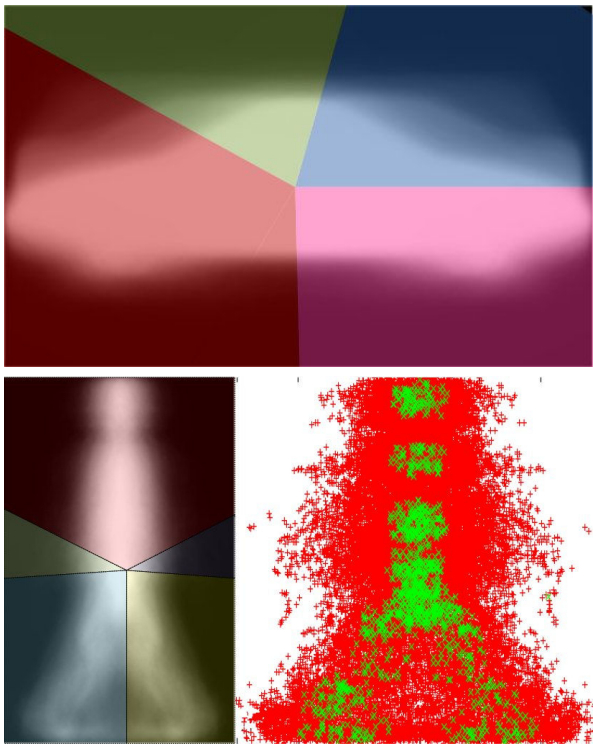

Fig. 6. **Top/Center**: Subparts, depicted as colored areas, and template masks, in white, both computed from the training set. Note that despite an unsupervised computation, the subparts exhibit some semantic interpretation. **Bottom**: Superfeatures are stable features in image and descriptor space. Shown are Shape Context descriptors at Hessian interest points (in red) for the class 'pedestrian'. The superfeatures are depicted in green.

relation between observations and labels. For the edge features, we compute the Euclidean distance between the corresponding 2D data points and a value based on the sum of distances from the decision boundary of AdaBoost, which is only high if both points are equally classified. The intuition here is that edges between equally labeled points reveal a higher likelihood of correct classification. To train the CRF we use L-BFGS gradient descent and for the inference we use max-product loopy belief propagation.

C. Results

Figure 7 shows some qualitative results of our detection and tracking algorithm. The data was collected from a tour of the SmartTer in the city of Zurich. It is particularly challenging due to occlusions, clutter and partial views. As one can see, cars and pedestrians are detected correctly, together with a correct estimation of their motion vectors. In a quantitative evaluation, we obtained an Equal-Error-Rate (equal precision and recall) of 68% for pedestrians and 75.7% for cars.

IV. Conclusion

In this paper, we gave an overview of the techniques we use for the two key tasks in autonomous driving in urban environments: ego-motion estimation and dynamic object detection. We showed the usefulness of our approaches on real data, acquired with our robotic platform SmartTer.

References

[1] M. Buehler, K. Iagnemma, and S. Singh, *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*, Springer Tracts in Advanced Robotics, ISBN: 9783642039904, 2009.

[2] R.I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[3] M. Bosse, R. Rikoski, R. Leonard, and S. Teller, "Vanishing points and 3d lines from omnidirectional video," in *ICIP02*, 2002, pp. III: 513–516.

[4] P. I. Corke, D. Strelow, and S. Singh, "Omnidirectional visual odometry for a planetary rover," in *IEEE Int. Conf. on Intell. Rob. and Systems (IROS)*, 2004.

[5] Maxime Lhuillier, "Automatic structure and motion using a catadioptric camera," in *IEEE Workshop on Omnidirectional Vision*, 2005.

[6] D. Nister, O. Naroditsky, and Bergen J., "Visual odometry for ground vehicle applications," *Journal of Field Robotics*, 2006.

[7] Roland Goecke, Akshay Asthana, Niklas Pettersson, and Lars Petersson, "Visual vehicle egomotion estimation using the fourier-mellin transform," in *IEEE Intelligent Vehicles Symposium*, 2007.

[8] J.P. Tardif, Y. Pavlidis, and K. Daniilidis, "Monocular visual odometry in urban environments using an omnidirectional camera," in *IEEE IROS'08*, 2008.

[9] M. J. Milford and G. Wyeth, "Single camera vision-only slam on a suburban road network," in *IEEE Int. Conf. on Robotics and Automation, ICRA'08*, 2008.

[10] D. Scaramuzza and R. Siegwart, "Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles," *IEEE Transactions on Robotics*, vol. 24, no. 5, October 2008.

[11] D. Scaramuzza, F. Fraundorfer, and R. Siegwart, "Real-time monocular visual odometry for on-road vehicles with 1-point ransac," in *IEEE International Conference on Robotics and Automation (ICRA 2009), Kobe, Japan, 16 May*, 2009.

[12] D. Scaramuzza, F. Fraundorfer, M. Pollefeys, and R. Siegwart, "Absolute scale in structure from motion from a single vehicle mounted camera by exploiting nonholonomic constraints," in *IEEE International Conference on Computer Vision (ICCV 2009), Kyoto*, October 2009.

[13] M. C. Deans, *Bearing-Only Localization and Mapping*, Ph.D. thesis, Carnegie Mellon Univ., 2002.

[14] A Davison, "Real-time simultaneous localisation and mapping with a single camera," in *International Conference on Computer Vision*, 2003.

[15] L. A. Clemente, A. J. Davison, I. Reid, J. Neira, and J. D. Tardos, "Mapping large loops with a single hand-held camera," in *Robotics Science and Systems*, 2007.

[16] T. Lemaire and S. Lacroix, "SLAM with panoramic vision," *Journal of Field Robotics*, vol. 24, no. 1-2, pp. 91–111, 2007.

[17] D. Nistér, "An efficient solution to the five-point relative pose problem," in *CVPR03*, 2003, pp. II: 195–202.

[18] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A toolbox for easy calibrating omnidirectional cameras," in *IEEE International Conference on Intelligent Robots and Systems (IROS 2006)*, oct 2006.

[19] D. Scaramuzza, "Ocamcalib toolbox: Omnidirectional camera calibration toolbox for matlab," 2006, Google for "*ocamcalib*".

[20] D. Scaramuzza, F. Fraundorfer, M. Pollefeys, and R. Siegwart, "Closing the loop in appearance-guided omnidirectional visual odometry by using vocabulary trees," *Robotics and Autonomous System Journal (Elsevier), 2009, TO APPEAR*.

[21] B. Leibe, E. Seemann, and B. Schiele, "Pedestrian detection in crowded scenes," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2005.

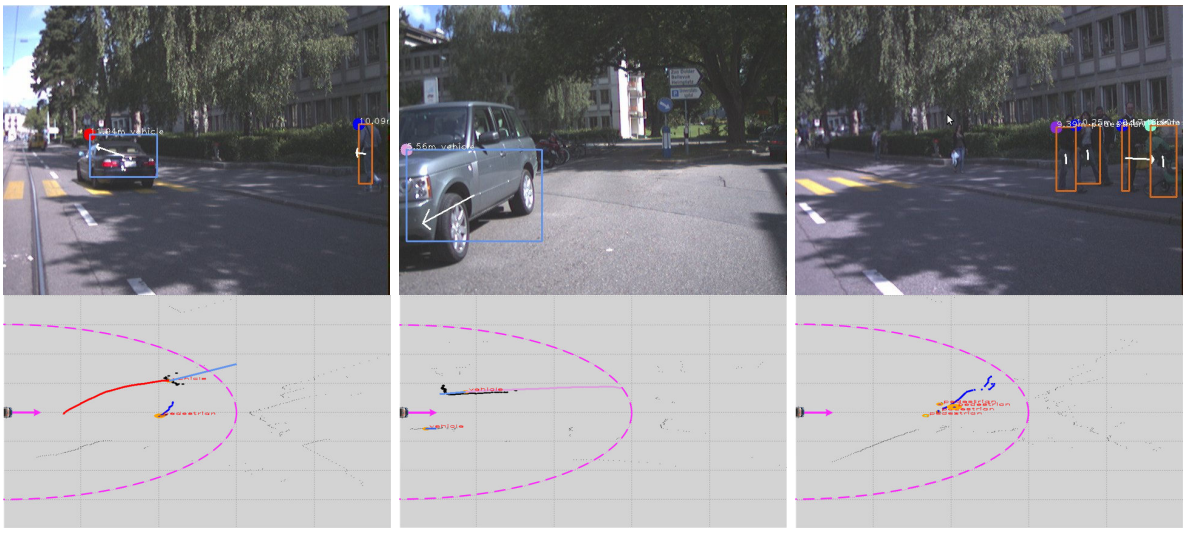

Fig. 7. Cars and pedestrian detected and tracked under occlusion, clutter and partial views. Blue boxes indicate car detections, orange boxes pedestrian detections. The colored circle on the upper left corner of each box is the track identifier. Tracks are shown in color in the second row and plotted with respect to the robot reference frame.

[22] L. Spinello, R. Triebel, and R. Siegwart, "Multiclass multimodal detection and tracking in urban environments," in *Int. Conference on Field and Service Robotics (FSR)*, 2009.

[23] L. Spinello, R. Triebel, and R. Siegwart, "Multimodal detection and tracking of pedestrians in urban environments with explicit ground plane extraction," in *IEEE Int. Conf. on Intell. Rob. and Systems (IROS)*, 2008.

[24] L. Spinello, R. Triebel, and R. Siegwart, "Multimodal people detection and tracking in crowded scenes," in *Proc. of the AAAI Conf. on Artificial Intelligence*, July 2008.

[25] M. Rufli and R. Siegwart, "On the application of the D* search algorithm to time-based planning on lattice graphs," in *Proc. of the European Conference on Mobile Robots*, 2009, to appear.

[26] Y. Freund and R.E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 5, 1997.