Proceedings of the IEEE ICRA 2009
Workshop on People Detection and Tracking
Kobe, Japan, May 2009

Improved Multi-Person Tracking with Active Occlusion Handling

A. Ess[1], K. Schindler[1,2], B. Leibe[3], L. van Gool[1,4]

[1] Computer Vision Laboratory, ETH Zurich, Switzerland
[2] Computer Science Dept., TU Darmstadt, Germany
[3] UMIC Research Centre, RWTH Aachen, Germany
[4] ESAT/PSI-VISICS IBBT, KU Leuven, Belgium

{aess|leibe|schindler|vangool}@vision.ee.ethz.ch

***Abstract*— We address the problem of vision-based multi-person tracking in busy inner-city locations using a stereo rig mounted on a mobile platform. Specifically, we are interested in the application of such a system for autonomous navigation and path planning. In such a scenario, semantic information about the moving scene objects becomes important. In order to estimate this robustly, we combine classical geometric world mapping with multi-person detection and tracking. In this paper, we refine an approach presented in earlier work, which jointly estimates camera position, stereo depth, object detections, and trajectories based only on visual information. We analyze the influence of the trajectory generator, which forms part of any tracking-by-detection system, and propose a set of measures to improve its performance. The extensions are experimentally evaluated on challenging, realistic video sequences recorded at busy inner-city locations. The results show that the proposed extensions significantly improve overall system performance, making the resulting detecting and tracking capabilities an interesting component of future navigation system for highly dynamic scenes.**

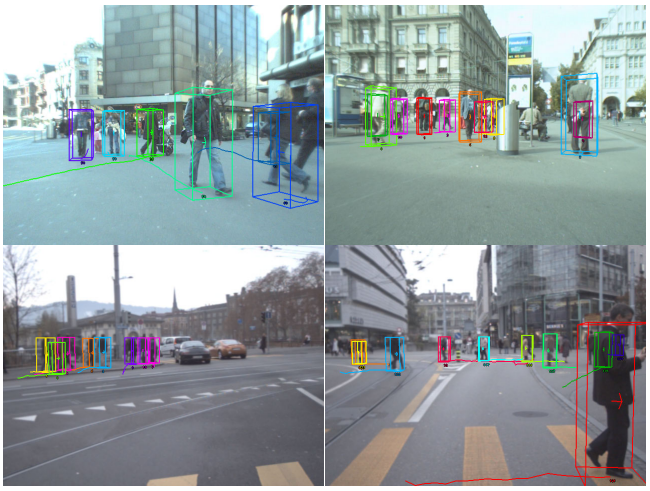


Fig. 1. Reliable tracking in busy urban scenarios requires careful design of trajectory (candidate) generation, accounting for partial occlusions, a multitude of scales, and measurement uncertainties.

I. Introduction

Reliable autonomous navigation of robots and cars requires appropriate models of their static and dynamic environment. While remarkable successes have been achieved in relatively clean highway traffic situations [3] and other largely pedestrian-free scenarios such as the DARPA Urban Challenge [7], scenes with many independently moving pedestrians, as in busy city centers, still pose significant challenges. What makes the task so much harder is the large number of independently moving actors that are frequently occluding each other. To represent such environments and make predictions for path planning, semantic information about the individual moving objects becomes a vital component.

Compared to range sensors such as LIDAR or SONAR, digital cameras offer the advantage that they deliver not only geometry, but also rich appearance information, which is more amenable to semantic interpretation. Recent work has shown that with modern computer vision tools, vision-based modeling of the environment for robot navigation is becoming possible [9], [27]. A key ingredient of these visual modeling approaches is that they partially rely on semantic *object category detection*—in the context of autonomous driving especially detection and tracking of cars and pedestrians.

For dynamic path planning, pedestrians need not only be detected, but should also be tracked over time in order to predict their future locations. However the two tasks are closely related: State-of-the-art approaches for people tracking in complex environments are based on the tracking-by-detection paradigm, in which the output of an (appearance based) object detector is linked between frames to recover pedestrian trajectories. In this work, we adopt such an approach for robust multi-person tracking and investigate some important design choices for improving its performance.

Our system is purely visual, using as input synchronized video streams from a forward-looking camera pair. Based on this data, the system continuously performs self-localization by visual odometry and obstacle detection using stereo depth and combines the resulting 3D measurements with tracking-by-detection, in order to follow pedestrians in the scene over time. Its results can be used directly as input for path planning algorithms which support dynamic obstacles. Key steps of our approach are the use of a state-of-the-art object detector for identifying an obstacle's category, as well as the reliance on a robust multi-hypothesis tracking framework to handle the complex data association problems that arise in crowded scenes. This allows our system to apply category-specific motion models for robust tracking and prediction. Our focus on vision alone does not preclude the use of other sensors such as LIDAR or GPS/INS—in any practical robotic system those sensors have their well-deserved place,

and their integration can be expected to further improve performance.

An important observation is that while each of the system components is affected by relatively strong noise, feedback between the components can remedy some of the resulting errors. Our system therefore has numerous feedback paths: we jointly estimate the ground surface and supporting object detections and let both steps benefit from each other; detections are transferred into world coordinates with the help of visual odometry and are grouped into 3D candidate trajectories by the tracker; selected tracks are then again fed back to stabilize visual odometry and depth computation through their predictions; finally, the results are combined in a dynamic occupancy map such as the one shown in Fig. 4(right), which allows free space computation for a later navigation module.

The main contribution of this paper is to investigate design options for the practical implementation of such a system and to evaluate their effects on overall performance. After reviewing related work (Sec. II) and the employed reconstruction and tracking system (Sec. III), we propose modifications to the trajectory generator (Sec. IV). The influence of these changes and of different stereo matching methods for depth computation are quantitatively evaluated in Sec. V.

II. Related Work

A main challenge in traffic scene understanding is to accurately detect moving objects in the scene. Such objects can be extracted independent of their category by modeling the shape of the road surface and treating everything that does not fit that model as an object (e.g. in [18], [24], [31]). However, such simple approaches break down in crowded situations where not enough of the ground may be visible. More accurate detections can be obtained by applying category-specific models, either directly on the camera images [6], [16], [23], [28], on the 3D depth information [1], or both in combination [10], [14], [25].

Tracking detected objects over time presents additional challenges due to the complexity of data association in crowded scenes. Targets are typically followed using classic tracking approaches such as Extended Kalman Filters (EKF), where data assignment is optimized using Multi-Hypothesis Tracking (MHT) [5], [20] or Joint Probabilistic Data Association Filters (JPDAF) [12]. Several robust approaches have been proposed based on those components either operating on depth measurements [21], [22], [26] or as tracking-by-detection approaches from purely visual input [13], [15], [17], [28], [30]. The approach employed in this paper is based on our own previous work [17]. It works online and simultaneously optimizes detection and trajectory estimation for multiple interacting objects and over long time windows by operating in a hypothesis selection framework.

III. System

Our vision system is designed for a mobile platform equipped with a pair of forward-looking cameras. From the synchronized videos, we estimate dense stereo depth, ground plane parameters, the platform's ego-motion, pedestrian tracks, and the locations of other (non-pedestrian) obstacles. Fig. 3(a) gives an overview of the proposed vision system. For each frame, the blocks are executed as follows. First, a depth map is calculated and the new frame's camera pose is predicted. Then objects are detected together with the supporting ground surface, taking advantage of appearance, depth, and previous trajectories. The output of this stage, along with predictions from the tracker, helps stabilize visual odometry, which updates the pose estimate for the platform and the detections, before running the tracker on these updated detections. As a final step, we use the estimated trajectories in order to predict future locations for dynamic objects and fuse this information with a static occupancy map. The whole system is held entirely causal, i.e. at any point in time it only uses information from the past and present.

Fig. 2. Mobile recording platforms used in our experiments. Note that in this paper we only employ image information from a stereo camera pair and do not make use of other sensors such as GPS or LIDAR.

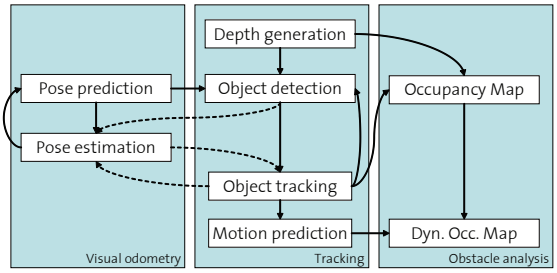
For the basic tracking-by-detection components, we rely on the framework described in [8], [9]. The main contribution of this paper is to propose a set of detailed improvements that considerably boosts tracking performance, both with respect to accuracy and speed, as explained in Section IV. The following subsections briefly review the overall system—see the above references for a full description.

A. Object Detection and Ground Plane Estimation

Instead of directly using the output of a pedestrian detector for the tracking stage, we introduce scene knowledge at an early stage to reduce false positives: a simple scene model is assumed where all objects of interest reside on a common ground plane. Instead of using a fixed ground plane, we allow a set of feasible planes to account for changes in terrain or tilted cameras due to e.g. braking. As a wrong estimate of this plane has far-reaching consequences for all later stages, we try to avoid making hard decisions here and instead model the coupling between object detections and the scene geometry probabilistically using a Bayesian network (see Fig. 3(b)). The network is constructed for each frame and models the dependencies between object hypotheses o_i, object depth d_i, and the ground plane $\boldsymbol{\pi}$ using evidence from the image $\mathcal{I}$, the depth map $\mathcal{D}$, a stereo self-occlusion map $\mathcal{O}$, and the ground plane evidence $\boldsymbol{\pi}_{\mathcal{D}}$ in the depth map. Following standard notation, the plate indicates repetition of the contained parts for the number of objects n.

An object's probability depends on its geometric world

(a)



(b)

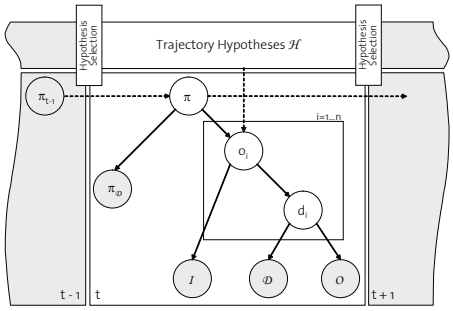


Fig. 3. Flow diagram for our vision system. (see text for details).

position and size, on its correspondence with the depth map, and on the object likelihood estimated by the object detector. The likelihood of each candidate ground plane is modeled by a robust estimator taking into account the uncertainty of the inlier depth points. The ground plane prior and the conditional probability tables are learned from training data.

In addition, we introduce temporal dependencies, indicated by the dashed arrows in Fig. 3(b). For the ground plane, we propagate the posterior from the previous frame, which stabilizes the per-frame information from the depth map. For the detections, we add a spatial prior for object locations that are supported by previously tracked candidate trajectories. As shown in Fig. 3(b), this dependency is not a first-order Markov chain, but reaches many frames into the past, as a consequence of the tracking framework explained in Section III-B.

The advantage of the Bayesian network formulation is that evidence is propagated in both directions: for a largely empty scene the ground plane can be reliably estimated from depth and significantly constrains object detection; in a crowded situation less of the ground is visible, but a large number of detected objects provide information about the ground plane.

B. Tracking and Prediction

Object detections from the previous step are placed into a common world coordinate system using camera positions estimated from visual odometry. The tracking system then uses detected object locations (projected onto the ground plane) as input for a multi-hypotheses tracker, similar to the one described in [17]: the set of object detections from the current and past frames is linked to an over-complete set of trajectory candidates with a holonomic constant-velocity model. Section IV deals with the careful design of the linking step—in this step the search space for the final set of pedestrian trajectories is generated, which obviously makes it important for system performance.

The set of candidate trajectories is then pruned to a minimal consistent explanation using model selection, while simultaneously resolving conflicts between overlapping candidates. In a nutshell, the pruning employs quadratic pseudo-boolean optimization to pick the subset of trajectories with maximal joint probability, given the observed evidence. This probability

- increases as the selected trajectories explain more detections and as they better fit the observed 3D locations and 2D appearance;
- decreases when trajectories would imply that two pedestrians occupy the same space at the same time;
- decreases with the number of required trajectories in order to balance the complexity of the model against its goodness-of-fit and to avoid over-fitting.

For the mathematical details, we refer to [17]. Important features of the method are automatic track initialization (usually, after ≈ 5 detections) and the ability to recover from temporary track loss and occlusion.

The selected trajectories are then used to provide a spatial prior for object detection in the next frame. This prediction has to take place in the world coordinate system, so tracking critically depends on an accurate ego-motion estimate.

C. Visual Odometry

To allow reasoning about object trajectories in world coordinates, the camera position for each frame is estimated using visual odometry. The employed approach builds upon previous work by [8], [19]. Please refer to those publications for details. Compared to standard visual odometry, our system includes scene knowledge obtained from the tracker to mask out image regions not showing the static background. Furthermore it explicitly detects failures by comparing the estimated position to a Kalman filter prediction. In the event of failure, the visual odometry is re-initialized to yield collision-free navigation (at the cost of possible global drift).

D. Static Obstacles

For static obstacles, we construct a stochastic occupancy map with the method from [2]: incoming depth maps are projected onto a polar grid on the ground and are fused with the integrated and transformed map from previous frames. Free space for driving is then computed with dynamic programming. In contrast to the original method, we filter out pedestrians found during tracking for two reasons: firstly, integrating non-static objects can result in smeared occupancy maps. Secondly, we are interested not so much in the *current* positions of the pedestrians as in their *future* locations. These can be predicted more accurately with a specific motion model inferred from the tracker.

IV. Trajectory Generation

Given space-time detections and a motion model, the obvious approach to generate putative trajectories is to continue the candidate trajectories from the previous frame

with an EKF. This method, which we refer to as *extension*, works quite robustly in cases without too much interaction between trajectories. To find newly appearing pedestrians and alternative explanations which contradict the previous candidates, one can additionally start an independent EKF backwards in time for each new detection, which we will call *parallel generation*. This basic approach was also used in our previous work [9], [17].

Here, we describe an ensemble of extensions to the hypothesis generation stage that (i) robustify data assignment, (ii) can actively handle occlusions from by both static as dynamic scene parts, and (iii) reduce the set of candidates and hence the runtime.

A. Clustering detections

When using detections from both cameras of a stereo pair, the same world object often generates one detection in each camera. Keeping two such detections separate increases the number of generated candidate trajectories, which increases the runtime, and can also affect the actual selection process. Hence, we propose to carry out a conservative clustering on detections from both cameras using world and appearance distance. This effectively replaces two measurements—originating from different views of the same object—by a single measurement for the physical 3D object. In our experiments, this reduces the number of candidates to ≈ 50–60% and the tracking time to $\approx 70\%$ of the original figures.

B. Greedy assignment

When generating/extending the candidate trajectories independently of each other, they cannot compete for measurements—the competition is left to the final selection algorithm. In difficult crowded cases, candidates will therefore include wrong measurements of other nearby objects. We have devised a simple strategy to remedy this behaviour: the *clustering* described above ensures that there is only one measurement per object. Hence, only the detection closest to the EKF's predicted location is used to update the state, rather than using all nearby detections weighted by the distance. In order to solve conflicts which arise when a measurement is the closest one for two or more candidate trajectories, the *extension* step is carried out simultaneously for all existing candidates, greedily assigning each detection to the trajectory candidate with the closest prediction. Candidates which do not manage to claim any detection during this process are merely extended through extrapolation. In the same way, only the best candidate at each time step is also chosen during *parallel generation*.

The effect of the competitive hard assignment of detections is twofold. Firstly, it avoids unwanted attraction between candidates and better separates closely interacting pedestrians. (When using soft assignment, the same measurement can influence several nearby trajectory candidates, pulling them closer together). Secondly, the set of candidates tends to be more compact, because each measurement can only support a single candidate in a crowded region, making weak candidates more prone to attrition.

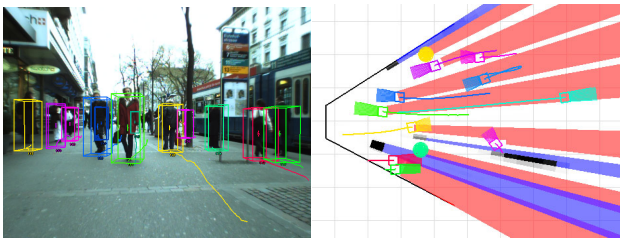


Fig. 4. From the image data (left) we infer occlusion regions (right) due to both static obstacles (black, casting blue umbra) and the previous frame's object predictions (red umbra). This information is used to correctly treat occluded candidate tracks.

C. Occlusion handling

Due to the camera placement on our vehicle, pedestrians frequently occlude each other, but are also often occluded by unmodeled scene objects. We therefore opt to explicitly model occlusion, rather than treat it as yet another case of missing detections. To this end, we generate an occlusion map on the ground plane, again discretized to a polar grid like the occupancy map in Section III-D. An example is shown in Fig. 4. The map contains the regions occluded by both pedestrians and static obstacles. To compute the map, pedestrian locations are estimated by extrapolating the previous tracker state to the current frame, whereas static obstacles are read out of the occupancy map.

As long as a candidate trajectory remains in an occluded region, it is kept alive and its state is extrapolated. Here the uncertainty modeling of the EKF becomes important: continued extrapolation without measurements leads to progressively larger location uncertainties and hence larger search regions for supporting detections. This increases the chances of finding the object once it becomes visible again. The *greedy assignment* described above meanwhile makes sure that such a candidate does not steal detections from less uncertain competitors. As a result, we obtain longer people tracks, which better support path planning [9].

V. Results

In order to evaluate our vision system, we applied it to two test sequences, showing strolls and drives through busy pedestrian zones. The sequences were acquired with the platforms seen in Fig. 2.[1] The first test sequence ("Seq. A"), recorded with platform (a) at considerably worse image contrast, contains 5'193 pedestrian annotations in 999 frames. The second test sequence ("Seq. B") consists of 800 frames and was recorded from a car passing through a crowded city center, where it had to stop a few times to let people pass. We annotated pedestrians in every fourth frame, resulting in 960 annotations for this sequence.

For a quantitative evaluation, we measure bounding box overlap in each frame and plot recall over false positives per image for three stages of our system. The results of this experiment are shown in Table I. We compare the raw

[1]Data and videos are available on `http://www.vision.ee.ethz.ch/~aess/icra2009/`.

detector output, the intermediate output of the Bayesian network, and the final tracking output. As can be seen, discarding detections that are not in accordance with the scene by the Bayesian network almost always increases recall at the same number of false positives. The tracking stage additionally improves the results and in most cases achieves a higher performance than the raw detector. It should be noted, though, that a single-frame comparison is not entirely fair here, since the tracker requires some detections to initialize (losing recall) and reports tracking results through occlusions (losing precision if the occluded persons are not annotated). However, the tracking stage provides the necessary temporal information that makes the entire motion prediction system at all possible. The line "Tracker (orig)" denotes the tracking performance of the system of [9] without the improvements described here. As can be seen, our new method consistently outperforms the original one. When only considering the immediate range up to 15m distance (which is suitable for a speed of 30 km/h in inner-city scenarios), performance is considerably better, as indicated by the second part of Table I.

Recall	Seq. A		Seq. B	
	FP 0.5	FP 1.0	FP 0.5	FP 1.0
Detector	0.57	0.65	0.61	0.67
Bayesian Net	0.65	0.67	0.63	0.66
Tracker (orig) [9]	0.60	0.74	0.52	0.60
Tracker (new)	0.64	0.73	0.55	0.65
Restricted to 15m				
Detector	0.51	0.62	0.76	0.78
Bayesian Net	0.66	0.66	0.74	0.74
Tracker (orig) [9]	0.72	0.74	0.70	0.70
Tracker (new)	0.73	0.77	0.80	0.80

Table I. Detection rates for two test sequences from different platforms. The Bayesian network consistently improves the detector. The tracker with the improvements proposed here also outperforms the original implementation [9]. Performance in the near range approaches a level where it becomes interesting for navigation.

We also compare the effect of using different methods for depth-map generation in Table II. This is of special interest, since nowadays a plethora of stereo algorithms of varying quality and runtime is available. Specifically, we compare the originally used belief-propagation-based stereo algorithm [11] (BP) with a fast GPU-based plane sweeping technique [4] (GPU), and a high-quality global-optimization approach [29]. Example depth maps are shown in Fig. 5. On the one hand, computationally intensive algorithms indeed yield an improvement in both scene analysis and tracking performance, but come at the cost of considerably higher runtime (20ms for GPU vs. 30s for the others). On the other hand, we are using robust statistics on the estimated depth values, hence top-of-the-line stereo matching does not yield noticeable improvements in system performance, despite producing visibly better depth maps.

	No depth		GPU		BP		Zach	
FP	0.5	1.0	0.5	1.0	0.5	1.0	0.5	1.0
BN	-	-	0.63	0.68	0.65	0.67	0.65	0.67
Tr.	0.19	0.29	0.60	0.70	0.64	0.73	0.64	0.73
Restricted to 15m								
BN	-	-	0.66	0.67	0.66	0.66	0.67	0.67
Tr.	0.32	0.47	0.66	0.74	0.73	0.77	0.73	0.78

Table II. Detection rates for Seq. A with different stereo matching methods. Better depth maps improve localization, and hence also tracking, in the near field. Fast GPU methods come at the expense of slightly worse performance. Since we use robust statistics on depth, elaborate stereo algorithms bring little improvement.

Fig. 6 shows results for Seq. A. The bounding boxes are color coded to show the tracked identities; darker boxes indicate objects in occlusion (due to the limited palette, some color labels repeat). Note that both adults and children are identified and tracked correctly even though they differ considerably in their appearance.

Fig. 7 demonstrates the system in an automotive application. Compared to the previous sequences, the viewpoint is quite different, and faster scene changes result in fewer data points for creating trajectories. Still, stable tracking performance can be obtained even for quite distant pedestrians.

VI. Conclusion

In this paper, we have presented a mobile vision system which combines classical geometric localization and mapping with tracking-by-detection of relevant object categories (in our case pedestrians). In this way, not only a geometric map of the world, but also tracks of dynamic objects of interest are available for subsequent path planning and decision making. Since object category detection inherently delivers the semantic information which type of object is tracked, customized motion models can be used for tracking and prediction.

The method relies on closely coupling the modules (detection, tracking, visual odometry, depth estimation). To resolve the complex interactions that occur between pedestrians in urban scenarios, a multi-hypothesis tracking approach is employed. The presented paper has focused on careful design of the hypothesis generation step, which turns out to be an important factor for improving system performance. The resulting system can handle very challenging scenes and delivers accurate predictions for many simultaneously tracked objects.

Acknowledgments. This project has been funded in parts by Toyota Motor Corporation/Toyota Motor Europe and the EU projects DIRAC (IST-027787) and EUROPA (ICT-2008-231888). We thank Nico Cornelis and Christopher Zach for providing GPU implementations of their stereo matching methods.

References

[1] K. O. Arras, O. M. Mozos, and W. Burgard. Using boosted features for the detection of people in 2d range data. In *ICRA*, 2007.

[2] H. Badino, U. Franke, and R. Mester. Free space computation using stochastic occupancy grids and dynamic programming. In *ICCV Workshop on Dynamical Vision (WDV)*, 2007.

[3] M. Betke, E. Haritaoglu, and L. S. Davis. Real-time multiple vehicle tracking from a moving vehicle. *MVA*, 12(2):69–83, 2000.

[4] N. Cornelis and L. van Gool. Real-time connectivity constrained depth map computation using programmable graphics hardware. In *CVPR (1)*, pages 1099–1104. IEEE Computer Society, 2005.

[5] I. J. Cox. A review of statistical data association techniques for motion correspondence. *IJCV*, 10(1):53–66, 1993.

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[7] DARPA. DARPA urban challenge rulebook. In *Webpage*, 2008. `http://www.darpa.mil/GRANDCHALLENGE/docs/Urban_Challenge_Rules_102707.pdf`.

[8] A. Ess, B. Leibe, K. Schindler, and L. van Gool. A mobile vision system for robust multi-person tracking. In *CVPR*, 2008.

Fig. 5. Example stereo depth maps for a given image. From left to right: GPU-based [4] (20 ms), belief-propagation based [11] (20–30 s), global optimization [29] (30–40 s) algorithm. Parts that are believed to be inaccurate (by a left-right check) are painted black. More advanced algorithms give visually better results, but take more time and are often not necessary (see text).

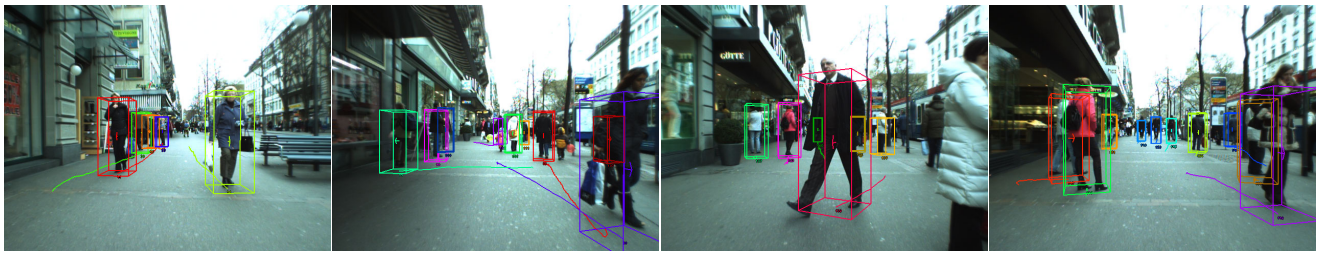


Fig. 6. Example tracking results for Seq. A, recorded from a child stroller.

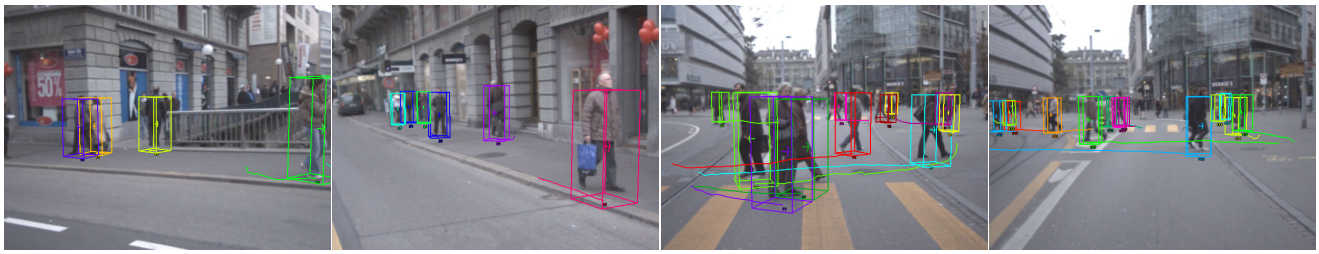


Fig. 7. Example tracking results for Seq. B, recorded from a moving car.

[9] A. Ess, B. Leibe, K. Schindler, and L. van Gool. Moving obstacle detection in highly dynamic scenes. In *ICRA*, 2009.

[10] A. Ess, B. Leibe, and L. van Gool. Depth and appearance for mobile scene analysis. In *ICCV*, 2007.

[11] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. *IJCV*, 70:41–54, 2006. Available from `http://people.cs.uchicago.edu/~pff/bp/`.

[12] T. E. Fortmann, Y. Bar Shalom, and M. Scheffe. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE J. Oceanic Engineering*, 8(3):173–184, 1983.

[13] D. Gavrila and S. Munder. Multi-cue pedestrian detection and tracking from a moving vehicle. *IJCV*, 73:41–59, 2007.

[14] D. Gavrila and V. Philomin. Real-time object detection for "smart" vehicles. In *ICCV*, pages 87–93, 1999.

[15] C. Huang, B. Wu, and R. Nevatia. Robust object tracking by hierarchical association of detection responses. In *ECCV*, 2008.

[16] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *IJCV*, 77(1-3):259–289, May 2008.

[17] B. Leibe, K. Schindler, N. Cornelis, and L. Van Gool. Coupled detection and tracking from static cameras and moving vehicles. *IEEE TPAMI*, 30(10):1683–1698, 2008.

[18] S. Nedevschi, R. Danescu, D. Frentiu, T. Graf, and R. Schmidt. High accuracy stereovision approach for obstacle detection on non-planar roads. In *Proc IEEE Intelligent Engineering Systems*, 2004.

[19] D. Nistér, O. Naroditsky, and J. R. Bergen. Visual odometry. In *CVPR*, 2004.

[20] D. B. Reid. An algorithm for tracking multiple targets. *IEEE T. Automatic Control*, 24(6):843–854, 1979.

[21] M. Scheutz, J. McRaven, and G. Cserey. Fast, reliable, adaptive, bimodal people tracking for indoor environments. In *IROS*, 2004.

[22] D. Schulz, W. Burgard, D. Fox, and A. Cremers. People tracking with mobile robots using sample-based joint probabilistic data association filters. *IJRR*, 22(2):99–116, 2003.

[23] A. Shashua, Y. Gdalyahu, and G. Hayun. Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. In *IVS*, 2004.

[24] M. Soga, T. Kato, M. Ohta, and Y. Ninomiya. Pedestrian detection with stereo vision. In *IEEE International Conf. on Data Engineering*, 2005.

[25] L. Spinello, R. Triebel, and R. Siegwart. Multimodal people detection and tracking in crowded scenes. In *Proc. of The AAAI Conference on Artificial Intelligence (Physically Grounded AI Track)*, July 2008.

[26] C.-C. Wang, C. Thorpe, and S. Thrun. Online simultaneous localization and mapping with detection and tracking of moving objects: Theory and results from a ground vehicle in crowded urban areas. In *ICRA*, 2003.

[27] C. Wojek and B. Schiele. A dynamic crf model for joint labeling of object and scene classes. In *ECCV*, 2008.

[28] B. Wu and R. Nevatia. Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet part detectors. *IJCV*, 75(2):247–266, 2007.

[29] C. Zach, J.-M. Frahm, and M. Niethammer. Continuous maximal flows and wulff shapes: Application to mrfs. In *CVPR*, 2009. accepted for.

[30] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008.

[31] L. Zhao and C. Thorpe. Stereo- and neural network-based pedestrian detection. In *ITS*, 2000.