Robust Multi-Person Tracking from a Mobile Platform

Andreas Ess, Bastian Leibe, Konrad Schindler, and Luc van Gool

***Abstract*— In this paper, we address the problem of multi-person tracking in busy pedestrian zones using a stereo rig mounted on a mobile platform. The complexity of the problem calls for an integrated solution that extracts as much visual information as possible and combines it through cognitive feedback cycles. We propose such an approach, which jointly estimates camera position, stereo depth, object detection, and tracking. The interplay between those components is represented by a graphical model. Since the model has to incorporate object-object interactions and temporal links to past frames, direct inference is intractable. We therefore propose a two-stage procedure: for each frame we first solve a simplified version of the model (disregarding interactions and temporal continuity) to estimate the scene geometry and an overcomplete set of object detections. Conditioned on these results, we then address object interactions, tracking, and prediction in a second step. The approach is experimentally evaluated on several long and difficult video sequences from busy inner-city locations. Our results show that the proposed integration makes it possible to deliver robust tracking performance in scenes of realistic complexity.**

***Index Terms*— Mobile vision, multi-object tracking, pedestrian detection, stereo depth, visual odometry, graphical model**

I. Introduction

Recent research successes have fostered the demand for mobile vision systems that can operate in unconstrained scenarios of daily human living. Building such systems has been a far-end goal of scene understanding since the 1970ies, but it is also a crucial requirement for many applications in the near future of mobile robotics and smart vehicles. So far, however, the sheer complexity of many real-world scenes has often stymied progress in this direction.

In this paper, we focus on the challenging task of multi-person tracking in busy street scenes as seen from a mobile observer. This could be a mobile robot, an electric wheelchair, or a car passing through a crowded city center. The scenario is extremely challenging due to a variety of factors: motion blur, varying lighting, large numbers of independently moving objects (sometimes covering almost the entire image), frequent partial occlusions between pedestrians, and sub-optimal camera placement dictated by constraints of a moving platform. (As the cameras are less than 1m above ground, a localization error of 1 pixel in y direction for an object 20m away equals about 1m in depth).

It has long been argued that scene analysis in such complex settings requires the combination of and careful interplay between several different vision modules. However, it is largely unclear how such a combination should be undertaken and which properties are critical for its success.

In this paper, we integrate visual odometry, depth estimation, pedestrian detection, and tracking in a graphical model and propose a two-step procedure to perform approximate inference with the model. An important component of the proposed integration is the concept of cognitive feedback. The underlying idea is that the higher-level information extracted by a certain vision module should be fed back to other modules in order to improve performance there, leading to cognitive loops. Several instantiations of this concept have been successfully demonstrated in recent years, among them the feedback from recognition to segmentation [7], [33], from geometry estimation to object detection [24], [29], from tracking to detection (*e.g.* [1], [19], [32], [39], [57]), and recently also feedback of object semantics to visual odometry [14].

In the described framework, data assignment problems arise both at the level of assigning image pixels to object detections and at the level of linking object detections to tracks. These ambiguities lead to implicit loops in the graphical model, which would require an infeasible modeling of the scene at the pixel level. Furthermore, the temporal connections to represent object persistence and the temporal continuity of geometric context over multiple frames would render the model prohibitively large. We therefore apply a hybrid approach, resolving only part of the modeled interactions through belief propagation and optimizing the remaining ones through a model selection procedure.

To make the model practically useful, inference is carried out in a causal way (*i.e.* each frame is treated separately, using only estimates from previous frames as fixed priors). Within each frame, depth measurements and object detections are jointly optimized disregarding object interactions; then these interactions are resolved and object trajectories are estimated using quadratic pseudo-boolean optimization. Finally, the camera motion estimate is updated using the current image and the estimated trajectories.

This paper makes the following main contributions: 1) We present an approach to simultaneously estimate scene geometry, detect objects, and track them over time in a challenging real-world scenario and from video input. This approach integrates and closely couples the different vision components in a combined system. 2) We demonstrate how this integration can be performed in a principled fashion, using a graphical model that allows depth measurements and object detections in each frame to benefit from each other and that links their results over time to object tracks with the help of visual odometry. 3) For inference in this model, we propose an iterative procedure that combines Belief Propagation and Quadratic Pseudo-Boolean Optimization to account for object-object interactions. 4) During the entire approach, we specifically address the question how to avoid system instabilities and guarantee robust performance. This is done by incorporating

A. Ess is with the Computer Vision Laboratory at ETH Zurich, Switzerland.

B. Leibe is with the UMIC Research Centre at RWTH Aachen University, Germany.

K. Schindler is with the Computer Science Department, TU Darmstadt, Germany.

L. van Gool is both with the Computer Vision Laboratory at ETH Zurich and with ESAT/PSI-VISICS IBBT at KU Leuven, Belgium.

automatic failure detection and correction mechanisms, which work together to avoid error amplification. 5) We experimentally validate the proposed method on challenging real-world data, encompassing over 5,000 video frame pairs, and demonstrate that the proposed integrated approach achieves robust multi-object tracking performance in very complex scenes.

The paper is structured as follows. After discussing related work in the following section, Section III showcases the components of the system. Section IV then presents the graphical model at the core of our approach which describes the dependencies between the different vision modules. Next, Section V introduces a two-step procedure for performing approximate inference on the model. This procedure consists of a simplified per-frame version of the model and separate optimization procedures for tracking over time and visual odometry. Practical considerations about robustness are presented in Section VI, before experimental results on a number of challenging video sequences are shown in Section VII. Section VIII concludes the paper with a summary and outlook.

II. Related Work

A. Visual Odometry

The majority of the work in visual odometry (VO) is based on local features and RANSAC-type hypothesize-and-test frameworks [38], [47]. Some other approaches include Hough-like methods [35] or recursive filtering [12], [13]. Most of these have however not been demonstrated on extended runs in realistic outdoor scenarios. The main problem with all these methods is the assumption that a dominant part of the scene changes only due to camera egomotion. As a result, these approaches are prone to failure in crowded scenes with many independently moving objects. While there has been work on multi-body Structure-from-Motion [34], [41], most systems are still constrained to short videos, and more importantly, assume sufficiently large, rigidly moving objects. In robotics, various approaches for SLAM in dynamic environments exist [5], [22], [55], related to the above, but mostly focusing on range data. In this paper, we propose to explicitly feed back information from object tracking to egomotion estimation, thereby introducing semantics.

B. Pedestrian Detection

Human detection has reached an impressive level [10], [16], [30], [53], [54], [57], with many systems also being able to estimate the silhouettes of the detected pedestrians [19], [30], [48], [58]. Still, pedestrian detection remains a difficult task due to large intra-category variability, scale changes, articulation, and frequent partial occlusion. To achieve robustness to adverse imaging conditions, the importance of *context* has been widely recognized. Depending on the authors, the rather loose notion of "context" can refer to different types of complementary information, including motion [11], [54], stereo depth [15], [19], scene geometry [24], [29], temporal continuity [1], [31], [32], [57], or semantics of other image regions [36], [40], [50], [51]. We build upon those ideas and extend them for our scenario.

C. Multi-object Tracking

Many approaches are available for multi-object tracking from stationary cameras (*e.g*. [4], [28]). The task is however made considerably harder when the camera itself moves. In such cases, background subtraction [49], [52] is no longer a viable option and tracking-by-detection appears to be the most promising alternative [1], [2], [19], [21], [29], [32], [39], [57], [60]. Targets are typically followed using classic tracking approaches, such as Extended Kalman Filters (EKF) [20], particle filters [25], or Mean-Shift tracking [8], which rely on a first-order Markov assumption and hence carry the danger of drifting away from the correct target. This danger can be reduced by optimizing data assignment and considering information over several time steps, as in Multi-Hypothesis Tracking (MHT) [9], [43] and Joint Probabilistic Data Association Filters (JPDAF) [18]. However, their combinatorial nature limits those approaches to consider either only few time steps [43] or only single trajectories over longer time windows [4], [26], [59]. Recently, [60] suggested a graph-based formulation for multi-target tracking that allows an efficient global solution even in complex situations. The approach operates on the entire video sequence and requires the detections for all frames as input. This precludes its online application to long sequences. In contrast, our approach works online and simultaneously optimizes detection and trajectory estimation for multiple interacting objects and over long time windows. For this, we build upon the hypothesize-and-test model selection framework from [31], [32] and extend it through the integration of stereo depth and visual odometry.

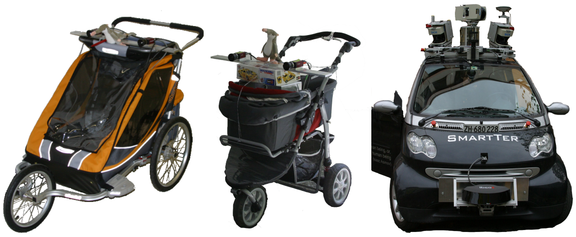


Fig. 1. Mobile recording platforms used in our experiments. Note that in this paper we only employ image information from a stero camera pair and do not make use of other sensors such as GPS or LIDAR.

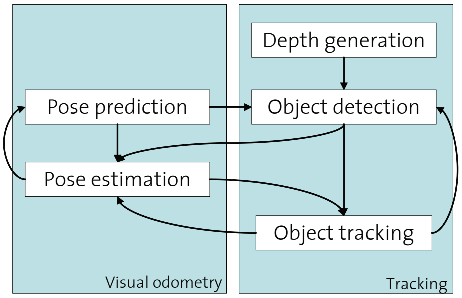


Fig. 2. Components of our mobile vision system and their connections, executed for each frame of a video sequence.

III. Overview

Our system is based on mobile platforms equipped with a pair of forward-looking cameras, as shown in Fig. 1. Under the predominantly occurring forward motion, the stereo setup is a better choice for self-localization than a monocular system because of the latter's weak geometric configuration [23]. Furthermore, generating depth maps has been well-studied for such setups [45], and dense depth information is of great help for constraining object detection and thus improving tracking and egomotion estimation.

Fig. 2 gives a schematic overview of the proposed vision system. This figure can be seen as the engineering view of the holistic graphical model we introduce in Section IV. As inference would be infeasible in a model of this size, we adopt a two-stage approach, as detailed below.

For each frame, the blocks are executed as indicated: first, a depth map is calculated and the new frame's camera pose is predicted. Then objects are detected. This step also encompasses the first stage of our graphical model that performs single-frame reasoning based on basic detector input, depth, and scene geometry. One of the novelties of this paper is to use the obtained information for stabilizing visual odometry, which then updates the pose estimate for the platform and the detections. Next, as second stage of our graphical model, the tracker is run on these updated detections.

The whole system is held entirely causal, *i.e.* at any point in time, we only use information from the past and present frame pairs. The following subsections detail the three main components of the system.

A. Object Detection

The graphical model we propose for tracking-by-detection is independent of a particular detector choice. In our experiments, we use three state-of-the-art detectors as basic input [10], [16], [30]. To obtain maximally possible recall, these detectors are applied with a low threshold. While this introduces a number of false positives, such errors are typically corrected by our integrated approach, as it considers scene geometry and introduces temporal dependencies.

B. Graphical Model for Tracking-by-Detection

This is the central part of our system. Designed as a holistic graphical model, it aims at combining the raw, independent detections from the image sequence to robust 3D trajectories with consistent identities. Due to the complexity of the represented interactions, the model is solved in a two-stage process. In a first stage, input from the basic detector is set in context with the rest of the scene using depth maps and assuming a ground plane. The effect of this is a set of reliable detections that adhere to the scene geometry and that can be placed in 3D camera coordinates. Camera position estimates from visual odometry are then used to transfer these detections into a common world coordinate frame, where the second stage of the graphical model combines tracking and occlusion reasoning based on a global optimization strategy. As the final step, the knowledge about tracked object locations in the image is used to improve visual odometry calculation for the next frame.

The most important effects of this are automatic track initialization (usually, after about 5 detections), as well as the ability to recover temporarily lost tracks, thus enabling the system to track through occlusions. Obviously, such a tracking system critically depends on an accurate and smooth egomotion estimate.

C. Visual Odometry

To allow reasoning about object trajectories in the world coordinate system, the camera position $\mathbf{P}$ for each frame is estimated using visual odometry. The employed system builds on previous work by [38]. In short, each incoming image is divided into a grid of 10×10 bins and an approximately uniform number of points is detected in each bin using a Harris corner detector with locally adaptive thresholds. The binning encourages a feature distribution suitable for stable localization. In the initial frame, stereo matching and triangulation provide a first estimate of the 3D structure. In subsequent frames, we use 3D-2D matching to get correspondences, followed by camera resection (3-point pose) with RANSAC [37]. Bundle adjustment is run on a sliding window of $n_b = 18$ past frames to polish the raw camera estimates. Older frames are discarded, along with points that are only supported by these removed frames.

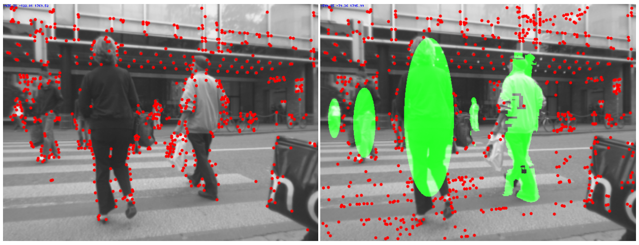


Fig. 3. Object detection and tracking give semantic meaning to the image and can be used to restrict localization efforts to image regions that are believed to contain static structures. (The algorithm for this step is described in detail in Section VI-B).

Important details for reliable performance are the use of 3D-2D matching to bridge temporally short occlusions of feature points and to filter out independently moving objects at an early stage, as well as a Kalman filter to predict the next camera position for feature detection (leading to a feature detection strategy similar to the "active search" paradigm in SLAM, *e.g.* [12]). Scene points are directly associated with a viewpoint-invariant SURF descriptor [3] that is adapted over time. In each frame, the 3D-2D correspondence search is then constrained by the predicted camera position. As mentioned above, only scene points without support in the past n_b frames are discarded. This allows one to bridge temporally short occlusions (*e.g.* from a person passing through the image) by re-detecting 3D points that carry information from multiple viewpoints and are therefore already reliably reconstructed.

For improved robustness, we introduce two measures: first, cognitive feedback from the tracker is used to constrain corner detection for visual odometry: the predictions delivered by the tracker identify image regions that are with a high probability occupied by moving objects (pedestrians), as shown in Fig. 3. No corners are extracted in these regions, which considerably reduces the number of incorrect matches (see Section VI-B). Second, we introduce an explicit failure detection mechanism, as described in [14]. In case of failure, the Kalman filter prediction is used instead of the measurement, all scene points are cleared, and the visual odometry is restarted from scratch. This allows us to keep the tracker running without resetting it. While such a procedure may introduce a small drift, a locally smooth trajectory is more important for our application.[1]

IV. Graphical Model

Fig. 4 shows the graphical model we assume for solving pedestrian detection and tracking from a mobile platform, an

[1] In fact, driftless global localization using only a moving camera rig is inherently impossible (except in retrospect in the case of loop closure). We believe that this capability, if needed, is best achieved by integrating other sensors, such as GPS and INS, as also argued *e.g.* in [61].

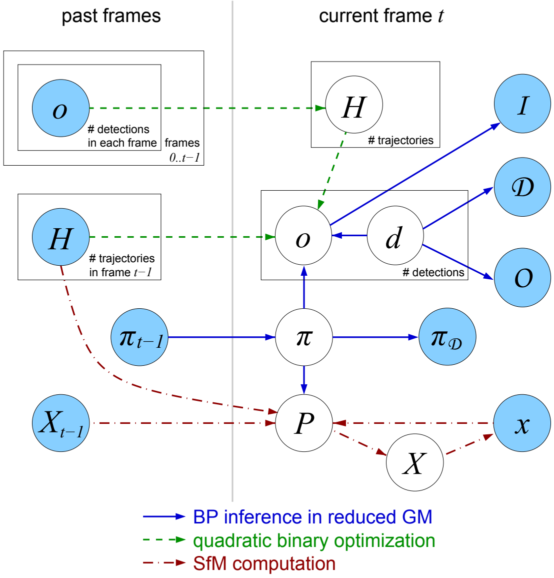


Fig. 4. The graphical model of our integrated system. See text for details.

Variable	Meaning
	Input
$\mathcal{I}$	Images of camera pair
$\mathcal{D}$	Depth maps of camera pair
$\mathcal{O}$	Occlusion maps inferred from depth maps
$\boldsymbol{\pi}_\mathcal{D}$	Ground-plane cue inferred from depth maps
$\mathbf{x}$	2D Harris corners (VO)
	Output / Hidden
o_i	Object hypotheses
d_i	Flag indicating validity of depth per object
H	Trajectory hypotheses
$\boldsymbol{\pi}$	Ground plane
$\mathbf{X}$	3D points (VO)
$\mathbf{P}$	Camera pose (VO)

Component	Model	Solution
Raw detector	Various	ISM, HOG, part-based
Reduced model for object detection	Bayesian network	Belief propagation
Reduced model for tracking	Minimum description length (MDL)	Multi-branch ascent
Visual odometry	Projective geometry	Structure from motion

TABLE I

TOP: VARIABLES USED THROUGHOUT THIS PAPER AND THEIR MEANING. BOTTOM: MATHEMATICAL MODELS ASSOCIATED WITH THE COMPONENTS.

overview of the employed variables, as well as mathematical models for the single components is given in Tab. I. The input consists of the sequence of images $\mathcal{I}$ from the stereo camera pair, together with their corresponding depth maps $\mathcal{D}$ and an occlusion map $\mathcal{O}$ specifying where the calculated depth can be trusted. From the same stereo depth information, we also calculate ground plane measurements $\boldsymbol{\pi}_\mathcal{D}$. Together, this information is used to infer object hypotheses o_i, object depth d_i, and the ground plane $\boldsymbol{\pi}$. Following standard graphical model notation [6], the plate indicates repetition of the contained parts for the number of objects n. In parallel, structure-from-motion (SfM) extracts 2D Harris corners $\mathbf{x}$, matches them to 3D points $\mathbf{X}$, and infers the camera pose $\mathbf{P}$. Together with the camera calibration from SfM and the estimated ground plane, detected objects are localized in 3D and grouped into trajectories H, which in turn exert a spatial prior on object locations o_i. Information about past detections $o_{i,t_0:t-1}$, trajectories $H_{j,t_0:t-1}$, 3D point locations $\mathbf{X}_{t-1}$, and the previous ground plane estimate $\boldsymbol{\pi}_{t-1}$ is propagated from past frames.

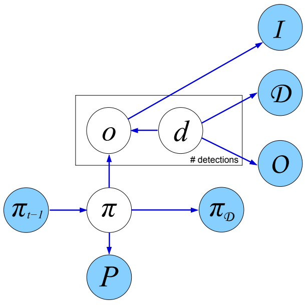


Fig. 5. The reduced graphical model for single-frame detection with additional information from past frames and depth maps.

To make inference tractable, we opt for a hybrid solution, where only part of the model is solved through probabilistic inference and where a global optimization stage is then used to select the best explanation for each frame. Thus, for each frame a reduced Bayesian network remains, which can be solved efficiently with Belief Propagation (blue arrows in Fig. 4). For each frame, object detections and the ground plane are estimated simultaneously in the reduced Bayesian network, taking advantage of appearance, depth, and trajectory information. The output, along with predictions from the tracker, helps stabilize visual odometry, which updates the pose estimate for the platform and the detections (red arrows in Fig. 4), before the tracker is run on these updated detections (green arrows in Fig. 4). The whole system is held entirely causal, *i.e.* at any point in time it only uses information from the past and present. In the following, we first describe the reduced Bayesian network for per-frame inference before putting it into context with the entire model.

A. *Reduced Model for Object Detection*

Fig. 5 shows the reduced Bayesian network we use for per-frame inference over object hypotheses o_i, object depth d_i, and the ground plane $\boldsymbol{\pi}$. Inference in this model is performed as follows:

$$P(\boldsymbol{\pi}, o_i, d_i, \mathcal{E}, \boldsymbol{\pi}_{t-1}) \propto P(\boldsymbol{\pi}_\mathcal{D}|\boldsymbol{\pi})P(\boldsymbol{\pi}|\boldsymbol{\pi}_{t-1})Q$$
$$Q = \prod_i P(o_i|\boldsymbol{\pi})P(o_i|d_i)P(d_i)P(\mathcal{I}|o_i)P(\mathcal{D}|d_i)P(\mathcal{O}|d_i)\ , \qquad (1)$$

where $\mathcal{E} = \{\mathcal{I}, \mathcal{D}, \mathcal{O}, \boldsymbol{\pi}_\mathcal{D}, \mathbf{P}\}$ is the evidence observed in the current frame and $\boldsymbol{\pi}_{t-1}$ indicates the ground plane from the previous time step. An object's probability depends both on its geometric world features (distance, size) $P(o_i|\boldsymbol{\pi})$ and its correspondence with the depth map (distance, assumption of uniform depth) $P(o_i|d_i)$. $P(\mathcal{I}|o_i)$ is the object probability estimated by the pedestrian detector (the time index t for the current frame was omitted for brevity — all variables without time index refer to time step t). Finally, we propagate the state of the ground plane in the previous frame as a prior $P(\boldsymbol{\pi}|\boldsymbol{\pi}_{t-1}) = Z((1-\alpha)P(\boldsymbol{\pi}) + \alpha P(\boldsymbol{\pi}_{t-1}))$, which augments the per-frame information from the depth map $P(\boldsymbol{\pi}_\mathcal{D}|\boldsymbol{\pi})$. Z is an appropriately chosen normalization constant.

In the following, the components of this Bayesian network are described in detail. All 3D calculations are executed in camera coordinates, *i.e.* the projection matrix is $\mathtt{P} = [\mathtt{K}|\mathbf{0}]$. This not only simplifies calculations and parameterizations, but it also keeps the set of possible ground planes in a range that can be trained in a meaningful way. For the subsequent tracking stage, the results are easily transferred into world coordinates with the camera orientation provided by visual odometry (Section III-C).

Ground Plane. As shown in previous publications [19], [24], [29], the ground plane helps substantially in constraining object detection to meaningful locations. It is defined in the current camera frame as $\boldsymbol{\pi} = (\mathbf{n}, \pi^{(4)})$, with the normal vector parameterized by spherical coordinates, $\mathbf{n}(\theta, \phi) = (\cos\theta\sin\phi, \sin\theta\sin\phi, \cos\phi)$. The ground plane parameters $\boldsymbol{\pi}$ are inferred from a combination of a prior from the previous frame, object bounding boxes, and the depth map evidence $\boldsymbol{\pi}_{\mathcal{D}}$, so that the system does not critically depend on any one individual cue. While accurate ground planes can be estimated directly from clean depth maps (see below), such methods break down in outlier-ridden scenarios. Thus, $\boldsymbol{\pi}_{\mathcal{D}}$ will just act as an additional cue in our Bayesian Network. Specifically, we consider the depth-weighted median residual between $\boldsymbol{\pi}$ and $\mathcal{D}$:

$$r(\boldsymbol{\pi}, \mathcal{D})^2 = \mathrm{med}_{\mathbf{x}\in\mathcal{D}} \, (\mathbf{n}^\top\mathbf{x} - \pi^{(4)})^\top \Sigma_{\mathcal{D}}^{-1} (\mathbf{n}^\top\mathbf{x} - \pi^{(4)}). \tag{2}$$

Here $\mathbf{x} \in \mathcal{D}$ denotes the set of 3D points inferred from $\mathcal{D}$, pruned according to the vehicle's maximally expected tilt angle and restricted to the lower part of the image for increased robustness to outliers. $\Sigma_{\mathcal{D}}$ accounts for the 3D point's uncertainty in the plane-to-point measurement. Given this robust estimate, we set[2]

$$P(\boldsymbol{\pi}_{\mathcal{D}}|\boldsymbol{\pi}) \propto e^{-r(\boldsymbol{\pi},\mathcal{D})^2}. \tag{3}$$

The prior $P(\boldsymbol{\pi})$ is also learned from a training set, as described in Section V.

Object Hypotheses. Object hypotheses $o_i = \{v_i, \mathbf{c}_i\}$, $(i = 1 \ldots n)$ are created from the output of a pedestrian detector[3] for each frame (typically, on the order of 10–100 detection hypotheses are used at each time step). They consist of a validity flag $v_i \in \{0, 1\}$ and a 2D center point with scale $\mathbf{c}_i = \{x, y, s\}$. Given a specific $\mathbf{c}$ and a standard object size (w, h) at scale $s = 1$, a bounding box can be constructed. From the box base point in homogeneous image coordinates $\mathbf{g} = (x, y + s\frac{h}{2}, 1)$, its counterpart in world coordinates is found as

$$\mathbf{G} = -\frac{\pi^{(4)}\mathtt{K}^{-1}\mathbf{g}}{\mathbf{n}^\top\mathtt{K}^{-1}\mathbf{g}}. \tag{4}$$

The object's depth is thus $z(o_i) = \|\mathbf{G}_i\|$. The box height $\mathbf{G}_i^h$ is obtained in a similar fashion. Because of the large localization uncertainty of appearance-based detection, the detector outputs for center and scale are only considered as estimates, denoted $\tilde{x}_i$, $\tilde{y}_i$, and $\tilde{s}_i$. Taking these directly may yield misaligned bounding boxes, which can in turn result in wrong estimates for distance and size. We therefore try to compensate for detection inaccuracies by considering a set of possible bounding boxes $\mathbf{b}_i^{\{k,\ell\}}$ for each o_i. These boxes are constructed from a set of possible real centers $\mathbf{c}_i = \{y_i, s_i\}$ (fixing $x_i = \tilde{x}_i$ due to its negligible influence), which are obtained by sampling around the detection, $y_i = \tilde{y}_i + k\sigma_y\tilde{s}_i$, $s_i = \tilde{s}_i + \ell\sigma_s\tilde{s}_i$. The number of samples, *i.e.* the range of $\{k, \ell\}$, is the same for every object. In the following, we omit the superscripts for readability. The object term is decomposed as

$$P(o_i|\boldsymbol{\pi}) = P(v_i|\mathbf{c}_i, \boldsymbol{\pi})P(\mathbf{c}_i|\boldsymbol{\pi}) \; . \tag{5}$$

By means of Eq. (4), $P(\mathbf{c}_i|\boldsymbol{\pi}) \propto P(\mathbf{G}_i^h)P(z(o_i))$ is expressed as the product of a distance prior $P(z(o_i))$ and of a size prior $P(\mathbf{G}_i^h)$ for the corresponding real-world object. We formulate the probability for a hypothesis' validity based on this, $P(v_i = 1|\mathbf{c}_i, \boldsymbol{\pi}) = \max_{k,l} P(\mathbf{c}_i|\boldsymbol{\pi})$.

Depth Map. The depth map $\mathcal{D}$ is a valuable asset for scene understanding that is readily available in a multi-camera system. However, stereo algorithms frequently fail, especially in untextured regions. For all our calculations, we therefore consider an additional *occlusion map* $\mathcal{O}$, which models the trust in each depth estimate based on a left-right check. This check computes the likelihood of an occlusion by comparing the color when remapping a pixel, as well as disparity estimates from the left and right camera. The intution behind the latter is that if a disparity estimate results from the estimator's smoothing, the left and right estimate will in most cases diverge. Using this consistency check, we integrate depth into our framework in a robust manner: each object hypothesis is augmented with a depth flag $d_i \in \{0, 1\}$, indicating whether the depth map for its bounding box is reliable $(d_i = 1)$ or not. As explained above, the depth term is decomposed into two parts:

$$P(o_i|d_i) = P(v_i|\mathbf{c}_i, d_i)P(\mathbf{c}_i|d_i). \tag{6}$$

First, we evaluate the stereo depth measured inside $\mathbf{b}_i$ and its consistency with the ground plane depth $z(o_i)$ as an indicator for $P(\mathbf{c}_i|d_i = 1)$. Second, we test the depth variation inside the box and define $P(v_i = 1|\mathbf{c}_i, d_i = 1)$ to reflect our expectation that the depth is largely uniform when a pedestrian is present. The measurements are defined as follows: the median depth inside a bounding box, $z(\mathcal{D}, \mathbf{b}_i) = \mathrm{med}_{\text{pixel }\mathbf{p}\in\mathbf{b}_i}\mathcal{D}(\mathbf{p})$, yields a robust estimate of the corresponding object's depth. Assuming additive white noise with covariance $\mathtt{C}_{\text{2D}}$ on pixel measurements, we find the variance $\sigma^2_{(z),i}$ of $z(\mathcal{D}, \mathbf{b}_i)$ using error backpropagation,

$$\mathtt{C}_i = \left(\mathtt{F}_i^{(1)\top}\mathtt{C}_{\text{2D}}^{-1}\mathtt{F}_i^{(1)} + \mathtt{F}_i^{(2)\top}\mathtt{C}_{\text{2D}}^{-1}\mathtt{F}_i^{(2)}\right)^{-1} \quad , \tag{7}$$

where $\mathtt{F}_i^{(j)}$ are the Jacobians of a projection using camera matrix j, thus $\sigma^2_{(z),i} = \mathtt{C}_i^{(3,3)}$. This yields

$$P_{(z),i}(x) \propto \mathcal{N}(x; z(\mathcal{D}, \mathbf{b}_i), \sigma^2_{(z),i}) \; . \tag{8}$$

$P_{(z),i}(x)$ thus models the probability that a given depth measurement x corresponds to the robustly estimated depth of the bounding box. For reasoning about depth uniformity, we consider the depth variation for all pixels $\mathbf{p}$ within $\mathbf{b}_i$, $V = \{\mathcal{D}(\mathbf{p}) - z(\mathcal{D}, \mathbf{b}_i)|\mathbf{p} \in \mathbf{b}_i\}$. To be robust against outliers, the estimate is restricted to the interquartile range $[LQ(V), UQ(V)]$, and depth uniformity is measured by the normalized count of pixels that fall within the confidence interval $\pm\sigma_{(z),i}$,

$$q_i = \frac{|\{x \in [LQ, UQ] | x^2 < \sigma^2_{(z)i}\}|}{UQ - LQ} \; . \tag{9}$$

This robust "depth inlier fraction" serves as basis for learning $P(v_i|\mathbf{c}_i, d_i = 1)$, as will be described in Section V-A. The probability $P(o_i|d_i = 0)$ is assumed uniform, since an inaccurate depth map gives no information about the object's presence.

[2] Note that since $r(\boldsymbol{\pi}, \mathcal{D})$ is obtained as a median, $P(\boldsymbol{\pi}_{\mathcal{D}}|\boldsymbol{\pi})$ should more appropriately be modeled as a Laplacian density. This made no difference in our experiments. We thank the unknown reviewer for pointing this out.

[3] Any standard detector can be plugged into our framework. In our experiments, we use three different publicly available methods [10], [16], [30].

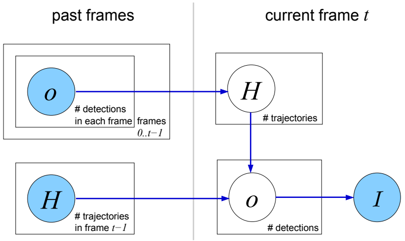


Fig. 6. Graphical model for trajectory estimation and detection prior.

We learn $P(d_i)$ from a training set based on the data from the occlusion map.

B. Reduced Model for Tracking

The aim of the tracking stage is to group detections into meaningful and physically plausible trajectories. In earlier work, we have introduced a hypothesize-and-test framework which uses model selection to jointly optimize object detection and trajectory estimation [31], [32]. Here, we follow this basic framework and adopt it for our application. Tracking is performed in world coordinates on the ground plane, so the aim of this step is to fit smooth trajectories to the detected object locations $\mathbf{x} = [x, z, t]^\top$ in a 3D spacetime volume. Fig. 6 shows the reduced graphical model for this step. This model contains two main types of interactions. On the one side, trajectory hypotheses $\{H_{j,t_0:t+1}\}$ are created from the set of stored detections of past frames $\{o_i\}_{t_0:t}$ in combination with new detection hypotheses from the current frame $\{o_{i,t+1}\}$. On the other side, trajectory hypotheses from past frames $H_{t_0:t}$ exert a spatial prior on certain object locations $o_{i,t+1}$, which raises the chance of finding detections there above a uniform background level $\mathcal{U}$. We model this prior as a Gaussian around the predicted object position using the trajectory's dynamic model $\mathcal{M}$. Thus,

$$p(o_{i,t+1}|\{H_{j,t_0:t}\}) \propto \max[\mathcal{U}, \max_j[P(x_{j,t+1})]], \qquad (10)$$

where $P(x_{j,t+1})$ is the normal distribution obtained from applying $\mathcal{M}$ to a hypothesis $H_{j,t_0:t}$.

A detection is specified by its position on the ground plane $\mathbf{x}_i$ and its color histogram $\mathbf{a}_i$, thus $o_i = \{\mathbf{x}_i, \mathbf{a}_i\}$. The probability that a detection o_i belongs to a given trajectory H_j depends on how well it fits the trajectory's dynamic motion model $\mathcal{M}$ and color model $\mathcal{A}$. In both cases, we use very simple models. For $\mathcal{M}$, we assume holonomic motion on the ground plane: a pedestrian is assumed to move with speed $v = |\mathbf{x}_t - \mathbf{x}_{t-1}|$ and direction $\theta = \arctan\frac{z_t - z_{t-1}}{x_t - x_{t-1}}$, and the uncertainty of the predicted position is modeled with an anisotropic Gaussian:

$$\mathcal{M} : \begin{cases} \mathbf{x}_{t+1} = \mathbf{x}_t + v[\cos\theta, \sin\theta]^\top \\ P(\mathbf{x}_{t+1}) \propto \mathcal{N}\big(\mathbf{x}_{t+1}, \Gamma_\theta \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix} \Gamma_\theta^\top\big) \end{cases}, \qquad (11)$$

with Γ_θ the rotation matrix and $\{\sigma_v, \sigma_\theta\}$ constant parameters of the model. As color model $\mathcal{A}$, we use the $8 \times 8 \times 8$ bin RGB histogram a_i of a detection (computed over an ellipse fitted inside the detected bounding box). The color model for a trajectory is the mean color histogram $\bar{\mathbf{a}}$ over all its detections.

We treat the dynamics and the appearance as independent: the probability that a given detection o_i at time $(t+1)$ belongs to a (partial) trajectory $H_{t_0:t}$ is given by

$$P(o_{i,t+1}|H_{t_0:t}) = p(o_{i,t+1}|\mathcal{A}(H_{t_0:t}))p(o_{i,t+1}|\mathcal{M}(H_{t_0:t})). \quad (12)$$

Tracking now consists of fitting a set of object trajectories $\{H_j\}$ with maximal joint probability $P(\{H_j\}|\{o_i\}) \propto P(\{o_i\}|\{H_j\})P(\{H_j\})$. Direct fitting is difficult due to the fact that the trajectories are not independent. The physical exclusion constraint demands that two trajectories shall not intersect in space-time ("no two people can occupy the same 3D space at the same time"). Moreover, following the principle of Occam's razor we prefer the simplest possible explanation for the observed detections, resulting in a prior $P(\{H_j\})$ which favors a smaller number of trajectories.

We incorporate both of those constraints by formulating the tracking problem in a hypothesize-and-test model selection framework [31]. Such an approach presupposes that we can sample a large set of potential candidates from the space of trajectories. Using Eq. (12), this can be done in the following way:

- Initialize the trajectory at an arbitrary detection and make a prediction (both forward and backward in time);
- Find detections at the new time step which support the trajectory by evaluating Eq. (12);
- Update $\{v, \theta, \bar{a}\}$, and iterate for the adjoining time steps.

In Section V-B, we will use this procedure to find a large number of candidate trajectories (in practice, even exhaustive sampling with only mild pruning is feasible), and we will select the jointly optimal subset with quadratic pseudo-boolean optimization.

V. Training and Inference

In this section, we describe how we perform training and inference in the described model. The system's parameters have been trained on a sequence (Seq. #1, see Section VII) with 490 frames, containing 1,578 annotations [4]. For learning the ground plane prior, we considered an additional 1,600 frames from a few selected environments with hardly any moving objects.

A. Object Detection

Belief Propagation. The graph of Fig. 5 is constructed per-frame, with all variables modeled as discrete entities and their conditional probability tables defined as described above. Inference is conducted using Pearl's Belief Propagation [42]. For efficiency reasons, the set of possible ground planes is pruned to the 20% most promising ones (according to prior and depth information).

Ground Plane. In input images with few objects, $\mathcal{D}$ can be used to directly infer the ground plane using Least-Median-of-Squares (LMedS) by means of Eq. (2),

$$\boldsymbol{\pi} = \min_{\boldsymbol{\pi}_i} r(\boldsymbol{\pi}_i, \mathcal{D}). \qquad (13)$$

Related but less general methods include *e.g.* the *v*-disparity analysis [27]. All such methods break down if less than 50% of the pixels in $\mathcal{D}$ support $\boldsymbol{\pi}$. For training, we use the estimate from Eq. (13), with bad estimates discarded manually.

For tractability, the ground plane parameters $(\theta, \phi, \pi^{(4)})$ are discretized into a $6 \times 6 \times 20$ grid, with bounds inferred from the training sequences. The discretization is chosen such that

[4] We have used data recorded at a resolution of 640×480 pixels (bayered) at 13-14 fps, with a camera baseline of 0.4 meters, respectively 0.6 meters for the car platform.

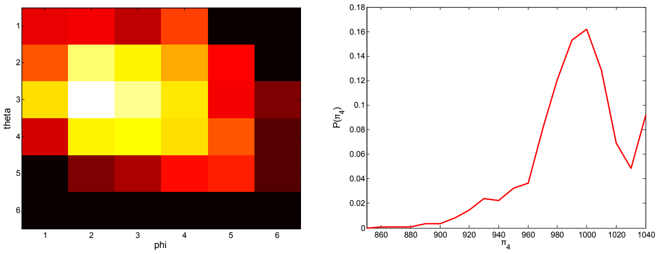


Fig. 7. Learned priors for (θ, ϕ) (left) and $\pi^{(4)}$ (right), projected onto $\pi^{(4)}$ and (θ, ϕ), respectively.

quantization errors are below 0.01 for θ and ϕ, resulting in component-wise abberations of maximally $5 \cdot 10^{-7}$ from the original $\mathbf{n}$. In our tests, the errors ensuing from the discretization of $\boldsymbol{\pi}$ were below 0.05 meters in depth for a pedestrian 15 meters away. Note that other choices of spherical coordinates for the normal vector would be better suited to the variability of the tilt angle. However, the described parametrization is sufficient, and alternative choices for discretization turn out to be more cumbersome because of switches from $-180°$ to $180°$. The training sequences also serve to construct the prior distribution $P(\boldsymbol{\pi})$. Fig. 7 visualizes $P(\boldsymbol{\pi})$ in two projections onto $\pi^{(4)}$ and (θ, ϕ).

Object Hypotheses. Object detections can be generated with any state-of-the-art pedestrian detector, parametrized in a conservative way so as to avoid false negatives as much as possible.

As the original detected locations $\tilde{x}, \tilde{y}, \tilde{s}$, and hence the bounding boxes, may not always be sufficiently accurate for reliable depth estimation, we model the offset between real and detected object centers by Gaussians (see [15] for details).

The object size distribution is chosen as in [24], $P(\mathbf{G}^h) \sim \mathcal{N}(1.7, 0.085^2)$ [m], though we consider different standard deviations σ_h in a first systematic experiment in Section VII. This is mainly to account for children and for the remaining discretization errors due to the sampling of $\mathbf{c}_i$. The depth distribution $P(z(o_i))$ is assumed uniform in the system's operating range of 0.5–30 m.

Depth Cues. The depth map $\mathcal{D}$ for each frame is obtained with a publicly available belief-propagation-based disparity estimation software [17]. See Fig. 8 for two example depth maps. The true distribution of $P(\mathbf{c}_i | d_i = 1)$ given the object's depth $z(o_i)$ and the depth map estimate $z(\mathcal{D}, \mathbf{b}_i)$ is very intricate to find. It involves many factors: first, the uncertainty of the object's center propagated to its distance. Due to the sampling of $\mathbf{c}_i$, we can neglect this factor. Second, it depends on $P_{(z),i}$ as defined in Eq. (8). Finally, using a fixed set of disparities introduces a quantization error, which is only to some extent covered by $P_{(z),i}$.

In Section VII, we compare two ways for modeling $P(\mathbf{c}_i | d_i = 1)$. The first option uses a non-parametric distribution $P(v_i | z(o_i) - z(\mathcal{D}, \mathbf{b}_i))$, learned from the training sequence. The second option models it using the dominating factor $P_{(z),i}(z(o_i))$ only.

For learning $P(v_i | \mathbf{c}_i, d_i = 1)$, we find the percentage q_i of pixels that can be considered uniform in depth for correct and incorrect bounding boxes using Eq. (9). As can be seen in Fig. 9, q_i is a good indicator of an object's presence. Using logistic regression, we fit a sigmoid to arrive at $P(v_i | \mathbf{c}_i, d_i = 1)$. In Section VII, we also test the use of $P(v_i = 1 | \mathbf{c}_i, d_i = 1) = \max_{k,l} P(\mathbf{c}_i | d_i = 1)$. With the same training set as above, we found $P(d_i = 1) \approx 0.96$.

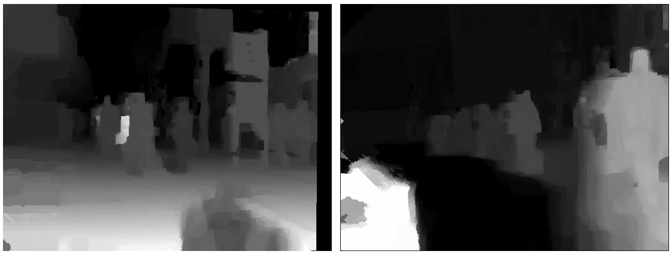


Fig. 8. Example depth maps. Most of the time, useful cues can be inferred (left), but robust measures have to account for faulty depth maps, *e.g.* missing ground plane (right).

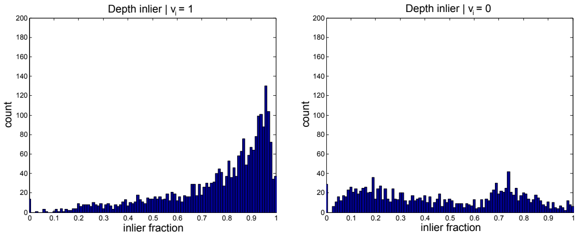


Fig. 9. Distribution of depth inliers for correct (left) and incorrect (right) detections, learned from 1,578 annotations and 1,478 negative examples. Based on those distributions, we learn a classifier using logistic regression.

Non-Maximum Suppression. In previous work [15], we have shown that it is possible to directly resolve the ambiguities from overlapping detections in the graphical model by a following Quadratic Binary Optimization stage. This procedure gave superior results for the ISM detector. Here, we adopt a simpler approach of just applying non-maximum suppression (NMS), so that the resulting framework can be readily combined with a wide variety of pedestrian detectors.

B. Tracking

For tracking, we employ a slightly adapted version of the multi-object tracking-by-detection framework from [31]. This approach applies model selection in order to find the set of trajectories that provides the best explanation for the observed evidence from past and present detections. This step is carried out by sampling a large, redundant set of candidate trajectories and pruning that set to an optimal subset. The candidate trajectories are not independent because of the twin constraints that two pedestrians cannot occupy the same location on the ground plane at the same time and that each object detection can only belong to a single pedestrian trajectory.

In our system, we generate the set of candidate trajectories by running the bi-directional trajectory-following method described in Section IV-B, starting from all detections within a large temporal window (for computational efficiency, the candidates from previous frames are cached and extended, and only those starting from new detections are generated from scratch). Each filter generates a candidate trajectory which obeys the physical motion constraints of a walking person and bridges short temporal gaps due to occlusion or detection failure. Note that candidates do *not* only originate from the accepted tracks of the last frame (like in classical trackers built on a first-order Markov assumption).

To select the jointly optimal subset of trajectories, we express the support (or utility) $\mathcal{S}$ of a trajectory $H_{t_0:t}$ by the evidence collected over its lifetime (the likelihood of the supporting detections) [31]:

$$
\begin{aligned}
\mathcal{S}(H_{t_0:t}|\mathcal{I}_{t_0:t}) &= \sum_i \mathcal{S}(H_{t_0:t}|o_{i,t_i})P(o_{i,t_i}|\mathcal{I}_{t_i}) \ , \\
&\propto P(H_{t_0:t}) \sum_i \mathcal{S}(o_{i,t_i}|H_{t_0:t})P(o_{i,t_i}|\mathcal{I}_{t_i}) \ ,
\end{aligned} \tag{14}
$$

$$
\mathcal{S}(o_{i,t_i}|H_{t_0:t}) = e^{-\lambda(t-t_i)} P(o_{i,t_i}|\mathcal{A}_{t_i}(H_{t_0:t}))P(o_{i,t_i}|\mathcal{M}_{t_i}(H_{t_0:t})). \tag{15}
$$

Choosing the best subset $\{H_j\}$ is now a model selection task and amounts to solving the quadratic binary problem

$$
\max_{\mathbf{m}} \left[\mathbf{m}^\top \mathtt{Q}\mathbf{m}\right] \ , \quad \mathbf{m} \in \{0,1\}^N \ , \tag{16}
$$

where $\mathbf{m}$ is an index vector, which specifies which candidates to use ($m_i = 1$) and which to discard ($m_i = 0$). The diagonal elements q_{ii} contain the individual likelihoods of candidate trajectory H_i, reduced by the "model penalty", a prior which favors solutions with few trajectories. The off-diagonal elements q_{ij} model the interaction between candidates i and j and contain the correction for double-counting detections consistent with both candidates, as well as a penalty proportional to the overlap of the two trajectories' footprints on the ground plane:

$$
\begin{aligned}
q_{ii} &= -\epsilon_1 G(H_{i,t_0:t}) + \\
&\quad + \sum_{o_{k,t_k} \in H_i} \big((1-\epsilon_2) + \epsilon_2(\mathcal{S}(o_{k,t_k}|\mathcal{I}_{t_k}) + \log P(o_{k,t_k}|H_i))\big) \\
q_{ij} &= -\frac{1}{2}\epsilon_3 O(H_i, H_j) - \\
&\quad - \frac{1}{2} \sum_{o_{k,t_k} \in H_i \cap H_j} \big((1-\epsilon_2) + \epsilon_2(\mathcal{S}(o_{k,t_k}|\mathcal{I}_{t_k}) + \log P(o_{k,t_k}|H_\ell))\big) \ ,
\end{aligned} \tag{17}
$$

where $H_\ell \in \{H_i, H_j\}$ denotes the weaker of the two trajectory hypotheses; $G(H_{t_0:t})$ is a model cost that penalizes holes in the trajectory; $O(H_i, H_j)$ measures the physical overlap between the footprints of H_i and H_j given average object dimensions; and $\epsilon_1, \epsilon_2, \epsilon_3$ are model parameters.

The maximization Eq. (16) is NP-hard, but there are several methods which find strong local maxima, *e.g.* the multi-branch method of [46], or QBPO-I [44]. The solution is an improved set of pedestrians for the current frame: most false detections are weeded out, since they usually do not have a supporting trajectory in the past (this is the main source of improvement), whereas missed detections are filled in by extrapolating those trajectories which have strong enough support in the previous frames. Typically, this step keeps between 25% and 35% of the candidate trajectories. In extreme cases, this figure extends to 8% and 100%, respectively. The ratio is strongly dependent on the complexity of the scene: the closer together the pedestrians move, the more candidates will be created. These are also the cases where a greedy maximization of Eq. (16) fails. When using the optimization method of [46], we however did not notice any problems with false local maxima. The selected tracks provide important information about the observed pedestrians and their motion through the scene.

Note that in theory, a better approximation can be achieved by iterating between tracking and object detection. In practice, this procedure converges after one additional iteration, without any effect on the final output, mostly due to the rather small differences between consecutive video frames.

VI. Improving Robustness

A. Failure detection

For systems to be deployed in real-life scenarios, failure detection is an often overlooked, but critical component. In our case, ignoring odometry failures can lead to erratic tracking behavior, since tracking relies on correct 3D world coordinates. As tracking is in turn used to constrain visual odometry, errors are potentially amplified further. Similarly, the feedback from object tracking as a spatial prior to detection can potentially lead to resonance effects if false detections are integrated into an increasing number of incorrect tracks. Finally, reliance on the ground plane to constrain object detection may lead to incorrect or missed detections if the ground plane is wrongly estimated. The proposed system relies on the close interplay between all components, so each of these failure modes could in the worst case lead to system instability and must be addressed.

Visual Odometry. To detect visual odometry failures, we consider two measures: firstly the deviation of the calculated camera position from the Kalman filter estimate and secondly the uncertainty (covariance) of the camera position. Thresholds can be set for both values according to the physical properties of the moving platform, *i.e.* its maximum speed and turn rate. Note that an evaluation of the covariance is only meaningful if based on rigid structures. Moving bodies with well distributed points could yield an equally small covariance, but for an incorrect position. When dynamic objects are disregarded, the covariance gives a reliable quality estimate for the feature distribution.

Object Tracking. The employed tracking method by construction accommodates failure detection and correction. Instead of taking a final decision at each time step and propagating only that decision to the next step, the approach builds upon a model selection framework to optimize tracks over a large temporal window. At each time instant, the tracking module explores a large number of concurrent track hypotheses in parallel and selects the most promising subset. This means that it can compensate for tracking errors and recover temporarily lost tracks.[5]

Object Detection and Ground Plane Estimation. These two components are kept stable by the continuous use of additional information from stereo depth. Depth measurements are employed both to support the ground plane estimate and to verify object detections. Thus, false predictions from the tracking system are corrected. Additionally, environments in which moving platforms can safely travel allow for a strong temporal prior $P(\boldsymbol{\pi}_{t-1})$ to smooth measurement noise.

B. Cognitive Feedback from Tracking to Visual Odometry

Besides failure detection, a key component for a robust system is the close interplay between the different modules. For detection and tracking, the graphical model provides a principled approach that by design implements cognitive loops between the two components. Even though the visual odometry is not directly part of this model, it can be integrated into this loop. In the following, we propose a feedback channel from detection-by-tracking to the

[5] The robustness comes at a cost: in hindsight, the optimal set of trajectories for a given time may change in the light of new evidence, similar to the MAP estimate of a particle filter. We therefore employ the method proposed in [31] in order to keep track of people identities.

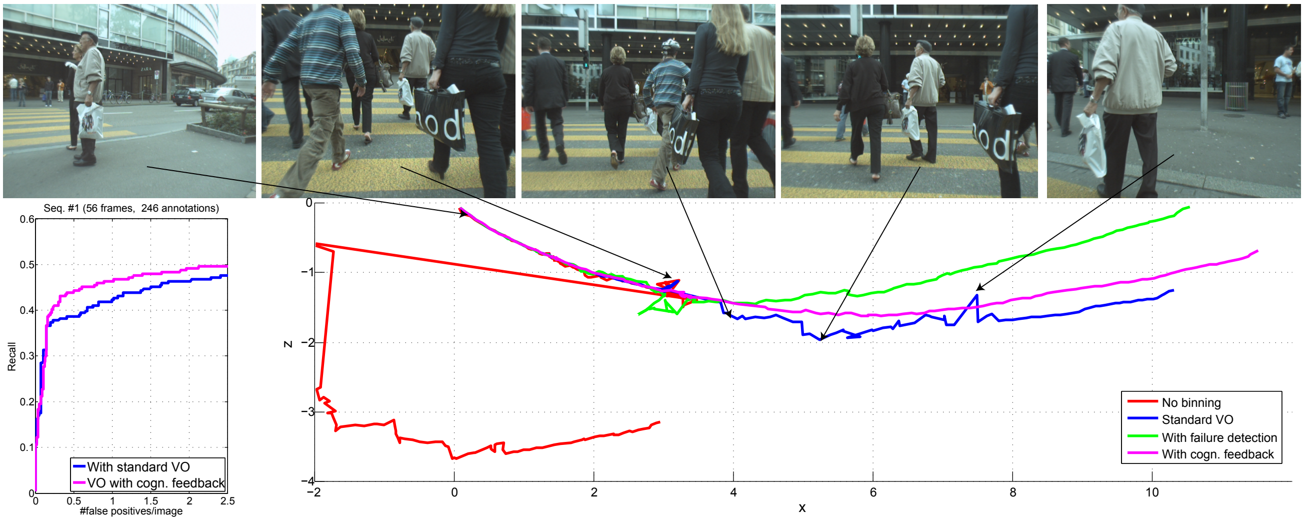


Fig. 10. Trajectory estimation of our system with and without cognitive feedback. (Top) A few frames of a difficult sequence. (Bottom) (Left) Recall/false positives for single detections with standard VO and VO using feedback. (Right) Trajectory estimates. As can be seen, the proposed feedback greatly stabilizes the egomotion estimation and leads to improved tracking performance. (Figure best viewed in color)

visual odometry system that increases the latter's robustness by introducing semantic information from the scene.

Standard algorithms for visual odometry (VO) assume a predominantly static scene, treating moving objects just the same as incorrect correspondences. Most systems use robust hypothesize-and-test frameworks such as RANSAC or Least-Median-of-Squares for removing such outliers. Recently, some multi-body Structure-from-Motion systems have been demonstrated on realistic video scenes [34]. However, those remain constrained to rigidly moving bodies such as cars and require a sufficient number of interest points for each model. We show that the use of basic scene understanding can effectively stabilize visual odometry by constraining localization efforts on regions that are likely to be part of the rigid scene.

In order to underline the importance of the proposed integration, consider the scene shown in Fig. 10, taken from one of our recordings. Here, our mobile platform arrives at a pedestrian crossing and waits for oncoming traffic to pass. Several other people are standing still in its field of view, allowing standard VO to lock onto features on their bodies. When the traffic light turns green, everybody starts to move at the same time, resulting in extreme clutter and blotting out most of the static background. Since most of the scene motion is consistent, VO fails catastrophically (as shown in the red curve). This is of course a worst-case scenario, but it is by no means an exotic case — on the contrary, situations like this will often occur in practical outdoor applications (we present another example in the results section).

Spatial binning for feature selection (as promoted in [38], [61]) improves the result in two respects: firstly, spatially better distributed features per se improve geometry estimation. Secondly, binning ensures that points are also sampled from less dominant background regions not covered by pedestrians. Still, the resulting path (shown in blue) contains several physically impossible jumps. Note here that a spike in the trajectory does not necessarily have to stem from that very frame. If many features on moving objects survive tracking (*e.g.* on a person's torso), RANSAC can easily be misled by those a few frames later. Failure detection using the Kalman filter and covariance analysis (in green) reduces spiking further, but is missing the semantic information that can prevent VO from attaching itself to moving bodies. Finally, the magenta line shows the result using our complete system, which succeeds in recovering a smooth trajectory. Detection performance improves as well (bottom row, left): when measuring recall over false positives per image (FPPI) on single detections, recall increases by 6% at 0.5 FPPI when using the cognitive feedback.

The intuition behind our proposed feedback procedure is to remove features on pedestrians using the output of the object tracker. For each tracked person, we mask out her/his projection in the image. If a detection is available for the person in the current frame, we use the confidence region returned by the object detector. If this region contains too large holes or if the person is not detected, we substitute an axis-aligned ellipse at the person's predicted position (this procedure is also employed for detectors that do not provide confidence maps). A few example masks are shown in Fig. 3. Note that this ellipse is not the same as the elliptical prior used for the temporal consistency in tracking: the one used here is directly fed back to the visual odometry in image coordinates, whereas in tracking, we use an elliptical prior in ground-plane world coordinates directly in the graphical model.

Given these object masks for a frame, the sampling of corners is adapted: thresholds are adapted to ensure a constant number of features, and corners lying on masked pixels are discarded. Even with imperfect segmentations, this approach improves localization by sampling the same number of feature points from regions where one is more likely to find rigid structure.

While this pedestrian crossing example represents a worst-case scenario for VO, the beneficial effect of the proposed cognitive feedback can also be seen in less extreme cases. For instance, for Seq. #2 (see Table II), estimated walking speed *before* Kalman filtering only spikes 15 instead of 39 times (in 1,200 frames) above a practical upper limit of 3 meters/second when using cognitive feedback. This means that the fallback options are used less frequently, and in turn that dead reckoning and hence introduction of drift are reduced. By optimizing the sampling

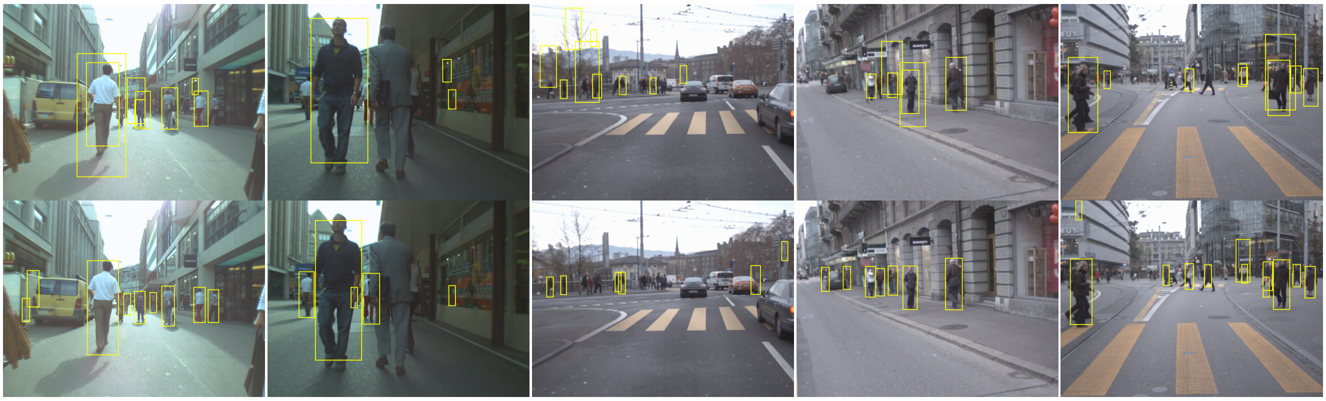


Fig. 11. Typical raw detections obtained by applying the detectors of Felzenszwalb *et al.* [16] (top row) and Dalal & Triggs [10] (bottom row) on Seq. #2 and Seq. #4. The detectors are applied out-of-the-box with a low confidence threshold on images that were rescaled to twice their original size in order to account for the detectors' rather large minimum scale.

Seq.	# Frames	Dist	VO Inliers Standard	VO Inliers w/ Feedback
#1	220	12m	30%	40%
#2	1,208	120m	27%	33%
#3	999	110m	39%	41%
#5	950	82m	40%	45%
#6	840	43m	12%	32%

TABLE II
OVERVIEW OF USED TEST SEQUENCES (FRAMES, APPROX. TRAVELLED DISTANCE), ALONG WITH AVERAGE PERCENTAGE OF VO INLIERS. THE COGNITIVE FEEDBACK CONSISTENTLY IMPROVES THE INLIER RATIO, ESPECIALLY IN HIGHLY DYNAMIC SCENES (#1,#6).

locations, the feedback generally improves the feature distribution and thus also the number of inliers. This can be seen in Table II for several test sequences (the other sequences will be introduced below).

VII. Experiments

In order to evaluate our mobile vision system, we applied it to five additional sequences, showing strolls or drives through busy pedestrian zones. In total, those sequences contain 5,017 frames, corresponding to more than 6 minutes. All sequences were acquired with the platforms seen in Fig. 2 and consist of two synchronized video streams recorded at 13–14 fps.[6]

The first test sequence ("Seq. #2") extends over 1,208 frames. We manually annotated all visible pedestrians $>$ 60 pixels in every fourth frame, resulting in 1,894 annotations. The second sequence ("Seq. #3") contains 5,193 annotations in 999 frames and considerably worse contrast. Both of those sequences were recorded with the child stroller setups shown in Fig. 1(left and middle). The third test sequence ("Seq. #4") has 800 frames and was recorded from a car passing through a crowded area, where it had to stop a few times to let people pass. The viewpoint is quite different, and faster scene changes result in fewer data points from which to estimate trajectories. Again, we annotated pedestrians in every fourth frame, resulting in 960 annotations. Finally, as a demonstration of the breaking points of our system, we show two very challenging sequences with fast turns ("Seq. #5") and an extreme number of moving pedestrians ("Seq. #6").

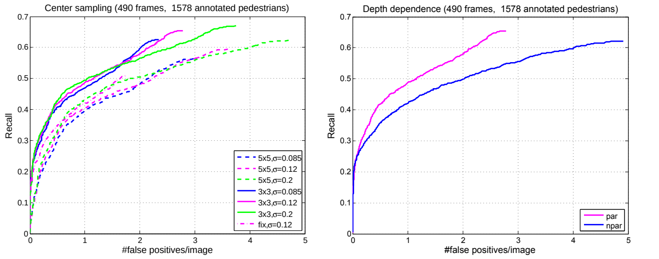


Fig. 12. Left: Influence of center/scale sampling and σ_h on performance. In all future experiments, we use 3×3 sampling and $\sigma_h = 0.12$. Right: Influence of depth term choice on performance, a parametric distribution performs better.

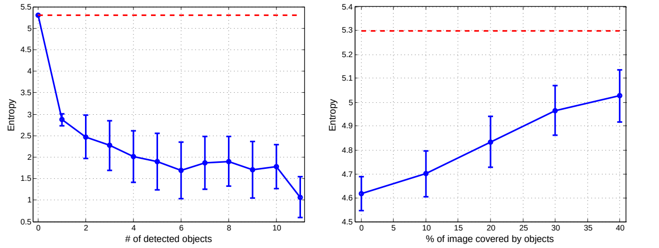
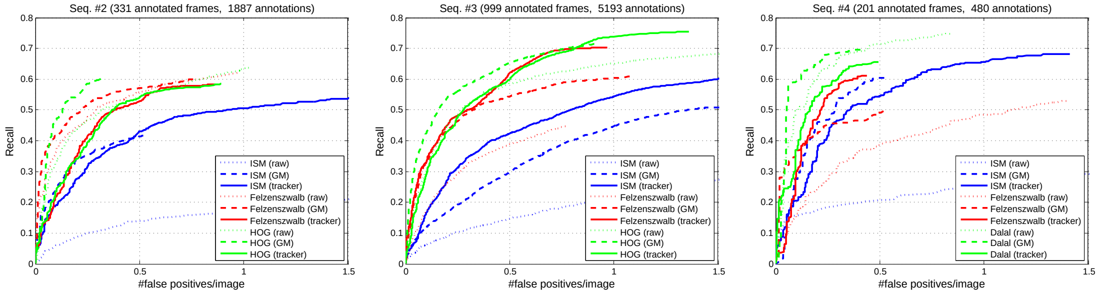


Fig. 13. Left: entropy of message from objects to ground plane as a function of the number of objects. Right: entropy of message from depth map to ground plane as a function of the image area covered by objects (and thus obstructing view on the ground plane).

We consider three different detectors in our experiments: the ISM detector by Leibe *et al.* [30], the HOG detector by Dalal & Triggs [10], and the recently proposed part-based HOG detector by Felzenszwalb *et al.* [16]. Since the latter two are constrained by their rather large minimum scale (96 pixels for [10]), we rescaled the input images to twice their original size for testing. Note, however, that this gives them a certain advantage over ISM in terms of performance. A few sample detections can be seen in Fig. 11.

For testing, all system parameters are kept the same throughout all sequences, except for setup-specific parameters such as camera calibration and height. Another exception is the ground plane prior for the car platform, which we assume to be Gaussian around the measured camera height. We measure performance by comparing

[6] All test sequences, including annotations, are available from `http://vision.ee.ethz.ch/~aess/`. We also provide all result videos at `http://vision.ee.ethz.ch/~aess/pami08`.



# persons	30
# trajectories	45
mostly hit	36
mostly missed	6
false alarms	4
ID switches	2

Fig. 14. (Left) Detection performance for the different detectors and system stages on Seq. #2, #3, and #4. (Right) Quantitative tracking results for part of Seq. #2. (See text for details. This figure is best viewed in color.)

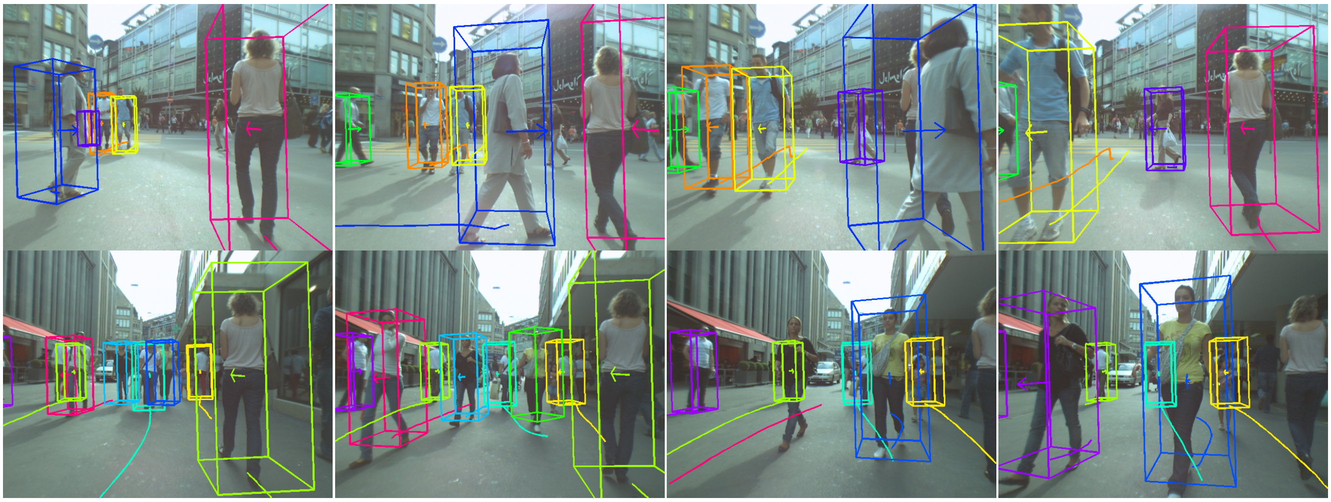


Fig. 15. Exemplary subsequences from Seq.#2. Note the long trajectories and the tracker's ability to handle temporary occlusions in complex scenarios.

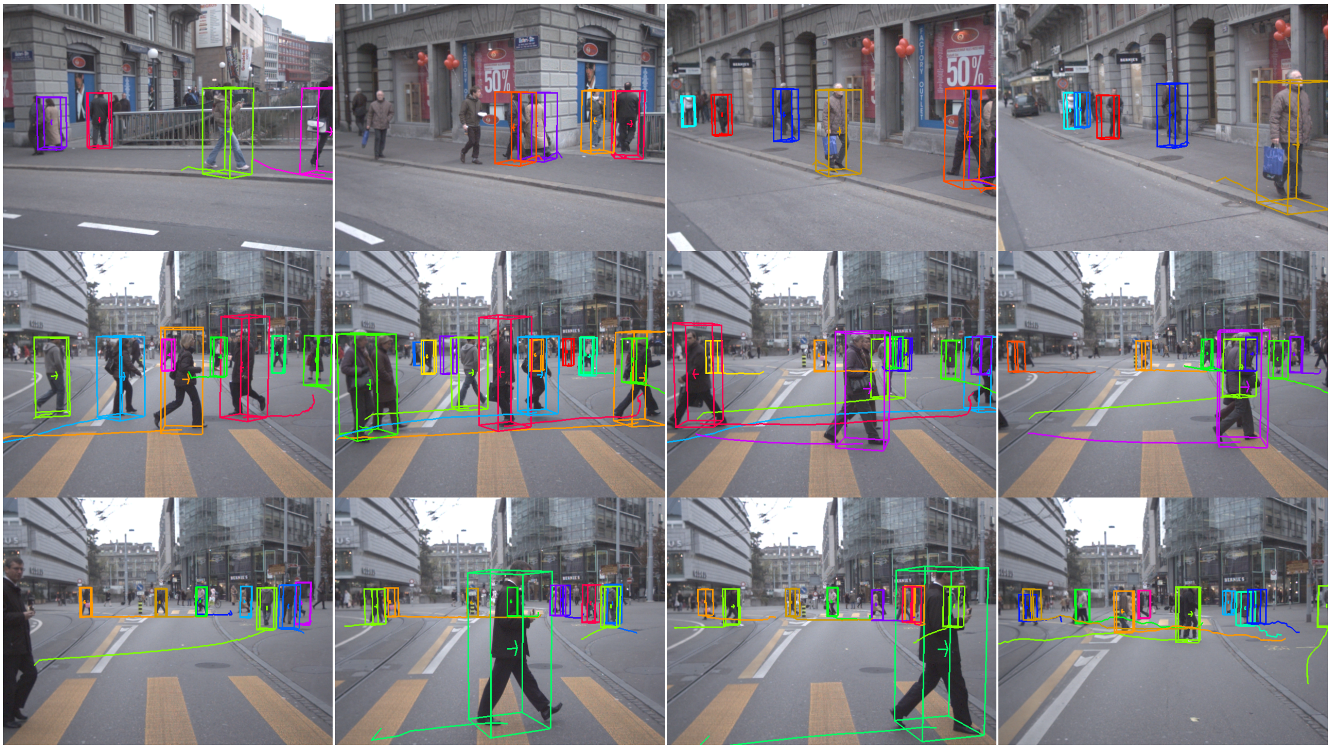


Fig. 16. Tracking results for Seq. #4. This sequence was recorded from a car driving through a crowded city center.

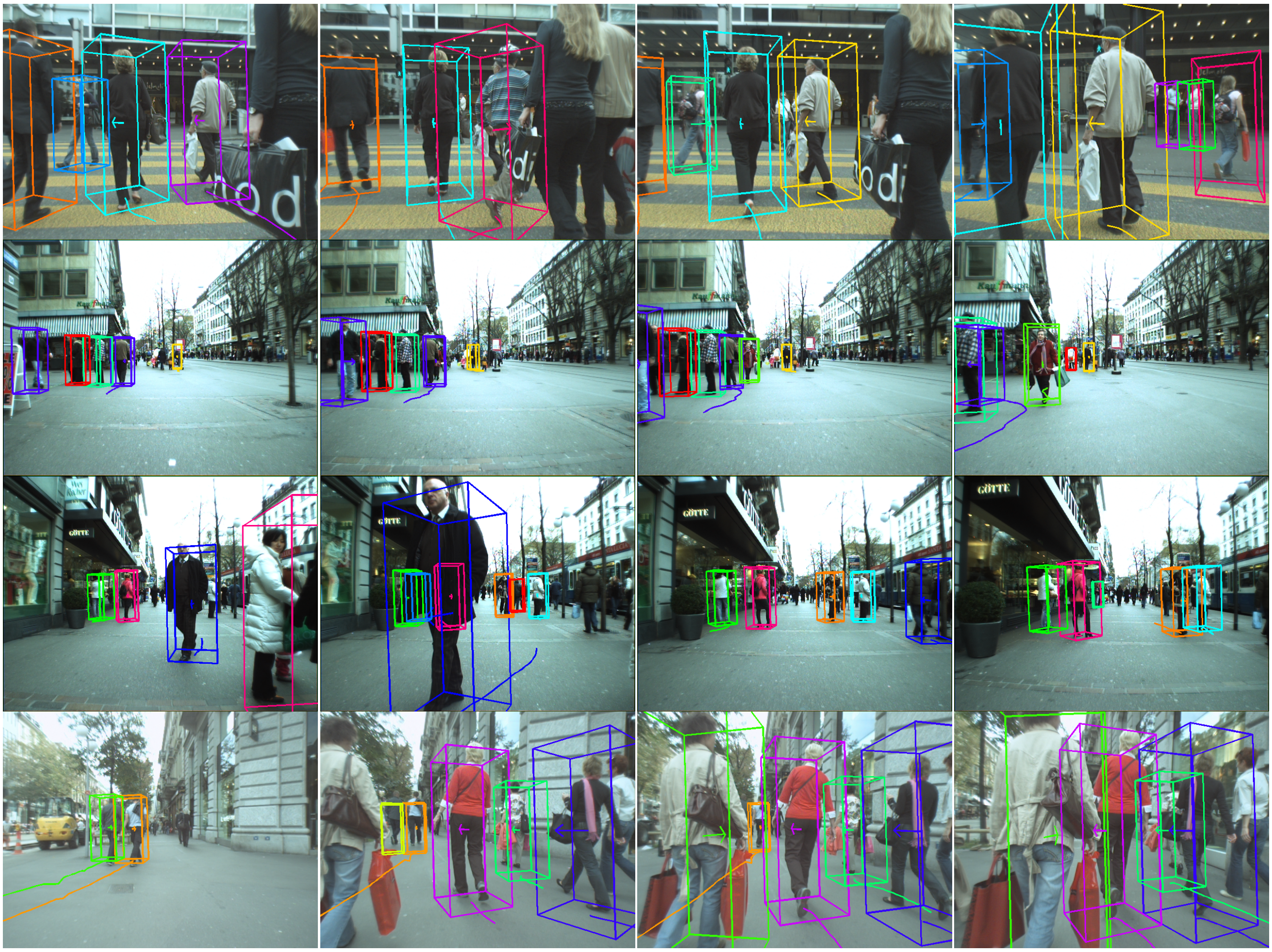


Fig. 17. Example tracking results for Seqs. #1, #3, and #6 showing strolls through busy pedestrian zones.

generated and annotated bounding boxes and plotting recall over false positives per image.

A. Validating the Model Parameters

The experiments in this section are performed on the training sequence using the ISM detector. They are used to determine the remaining parameters of the graphical model before it is applied to the test sequences.

First, we consider the standard deviation σ_h of the size prior, along with the sampling range $\{k, \ell\}$ in which the graphical model can shift the object center location $\mathbf{c}_i$. We consider no sampling, 3×3 ($k, \ell \in \{-1, 0, 1\}$), and 5×5 ($k, \ell \in \{-1, -0.5, 0, 0.5, 1\}$) sampling. Fig. 12(left) shows the resulting detection performance. As expected, a higher σ_h yields better precision at first, but recall grows too slowly. Due to the increased number of choices in Belief Propagation, the use of 5×5 sampling steps has also a negative effect on the performance. By just fixing the object center, recall is limited, as the algorithm cannot compensate for misaligned bounding boxes. A 3×3 sampling with $\sigma_h = 0.12$ thus seems a good compromise.

Second, we experimentally establish how to integrate the depth cues into our system. For $P(\mathbf{c}_i|d_i = 1)$, we consider either the learned non-parametric distribution $P(v_i|z(o_i) - z(\mathcal{D}, \mathbf{b}_i))$ ("npar") or a normal distribution inferred from Eq. (7) ("par"). As can be seen from the result plot (Fig. 12(right)), the non-parametric distribution for $P(\mathbf{c}_i|d_i = 1)$ performs worse. This is mostly due to a relatively small number of samples (especially at larger depths) for creating the necessary tables, as well as to a bias introduced by annotations and the training ground plane.

Our probabilistic approach to ground plane estimation was motivated by the idea that stereo depth based ground plane estimation and object detection can compensate for each other's weaknesses. In order to verify if this is indeed the case, we present the following experiment. In Fig. 13(left), we measure the entropy of the incoming messages from the objects to the ground plane node. As can be seen, the larger the number of objects, the lower the entropy, *i.e.*, the presence of many objects constrains the ground plane in a meaningful way. On the other hand, when there are hardly any objects, most of the depth map will contain evidence for the ground plane and will thus constrain it well. This is reflected in Fig. 13(right): the more image area is covered by objects, the less is covered by the ground plane. Thus, the entropy of the message from depth map to ground plane gets higher, as almost all ground planes become equally likely (in this case, a uniform distribution corresponds to an entropy of 5.3, indicated by the red dotted line in the plots).

B. Experimental Evaluation

Fig. 14 compares the single-frame performance for Seqs. #2, #3, and #4. For all 3 detectors, we plot their raw output ("raw"), the intermediate result obtained using the reduced graphical model ("GM"), as well as the final tracker output ("tracker"). In general, the HOG detector [10] gives the best results in terms of raw detections, with the part-based model of Felzenszwalb [16] a close second. ISM detection performance is slightly worse, mostly due to the fact that is was not run on the double image size and to its preference for side-views, which are rather rare in these sequences. As expected, the output of the graphical model considerably reduces the number of false positives by introducing scene knowledge, regardless of the raw detection input. Maximally reachable recall is hardly affected, *i.e.* the model only seldomly discards correct detections. The complete system also consistently ranks higher than the raw detector output. However, compared to the intermediate stage, its performance depends on the scene content. This is due to the nature of the tracker, which needs a few frames before initializing a track (losing recall) and which also reports currently occluded hypotheses (increasing false positives). Thus, depending on the complexity of the original scene (number of occlusions), the annotator (what is considered an occlusion?), and the number of missed detections on the basic level (the only case where the tracker can make up for in recall), the performance varies. A bounding-box level comparison is thus not favorable for the tracker.

We therefore also evaluated tracking performance manually in 450 frames of Seq. #2 using similar criteria as described in [57] (Tab. 14). We consider the number of pedestrians, the number of trajectories (if a pedestrian is occluded for > 10 frames, we count a new trajectory), the number of mostly hit trajectories ($>80\%$ covered), mostly missed trajectories ($<20\%$ covered), the number of false alarms, and the number of ID switches (meaning the tracker drifts from one person to another). On average, 75% of a trajectory are covered by the tracker. The missed trajectories belong mostly to pedestrians at smaller scales and to two children that do not fit the size prior.

For Seq. #3, the authors of [60] report 70% recall at 1 FPPI, again with a bounding-box level evaluation. While they do not use stereo data, their approach is a batch process (requiring the detections of the entire video sequence) and explicitly handles occlusion. Using the HOG detector, our system performs comparably with 74.4% recall at 1 FPPI.

Example tracking results for Seq. #2 are shown in the first two rows of Fig. 15. Our system's ability to track through occlusion is demonstrated in the top row: please note how the woman entering from the left temporarily occludes almost every part of the image. Still, the tracker manages to pick up the trajectory of the woman on the right again (in red). Results for Seq. #4 can be seen in Fig. 16. This sequence is considerably harder, as the different heights for sidewalk and street violate the flatness assumption for the ground plane. Furthermore, the higher viewpoint brings more people into view. As can be seen, the system manages to reliably track people both under fast egomotion and through considerable occlusions when standing at pedestrian crossings.

Finally, Fig. 17 shows additional tracking results for Seqs.#1, #3, and #6. Again, our system manages to produce long and stable tracks in complex scenarios with a considerable degree of occlusion. In the second row, a pedestrian gets successfully tracked on his way around a few standing people, and two pedestrians are detected at far distances. The third row again demonstrates tracking through major occlusion. Finally, the bottom row shows an example scenario from Seq. #6 with many pedestrians blocking the camera's field-of-view. As mentioned above, scenes of this complexity are at the limit of what is currently possible with our system.

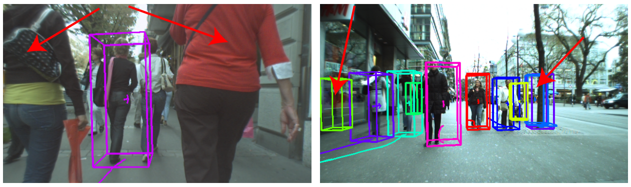


Fig. 18. Typical false negatives (large scale) and false positives (reflections, trees).

C. Runtime Performance

Apart from the detectors, the entire system is integrated in C/C++, with several procedures taking advantage of GPU processing. By substituting the Belief Propagation based stereo depth estimation by a fast GPU approximation, we can achieve processing times of around 300 ms per frame on an Intel Core2 CPU 6700, 2.66GHz, nVidia GeForce 8800 at essentially the same system accuracy. While the current bottleneck is the detector stage (all of the tested detectors were run offline and needed about 30 seconds per image), we want to point out that for the HOG detector, GPU implementations exist [56], which have the potential to remove this bottleneck.

VIII. Conclusion

In this paper, we have presented an integrated system for multi-person tracking from a mobile platform. The different modules (here, appearance-based object detection, depth estimation, tracking, and visual odometry) were integrated in a graphical model and exchanged information using a set of feedback channels. This close coupling proved to be a key factor in improving system performance. We showed that special care has to be taken to prevent system instabilities caused by erroneous feedback. Therefore, a set of failure prevention, detection, and recovery mechanisms was proposed. In future work, we plan to investigate whether it is feasible to apply a control-theoretic approach in order to handle those components in a unifying framework. This will however require modeling the non-linearities of third-party components and catering for the different platforms.

As our experimental evaluation shows, the resulting system can handle very challenging scenes and track many interacting pedestrians simultaneously and over long time frames. Finally, we demonstrated that the entire system can be efficiently implemented. As not all speedup possibilities are explored yet, the current runtime of 300 ms per frame raises hopes that practical online experiments in real vehicles will not be too far away anymore.

In future work, we will try to improve the individual components further, both with respect to speed and performance. For instance, very close pedestrians, for which only part of the torso is visible, are often missed by the pedestrian detector, as shown in Fig. 18. A graceful degradation in form of image-based tracking might be a possibility to prevent system breakdown in such cases. Further work is also required to address typical detection failures, such as false positives on trees or reflections and missing detections at too large or small scales. In addition, we plan to take advantage of depth information in order to detect other kinds of (static and dynamic) obstacles in the vehicle's path. Finally, further combinations with other modules, such as world knowledge inferred *e.g.* from map services, provide other exciting feedback possibilities that we plan to investigate in the future.

Acknowledgments. This research has been funded, in parts, by Toyota Motor Corporation/Toyota Motor Europe and the EU projects DIRAC (IST-027787) and HERMES (IST-027110).

References

[1] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *CVPR*, 2008.

[2] S. Avidan. Ensemble tracking. In *CVPR*, 2005.

[3] H. Bay, A. Ess, T. Tuytelaars, and L. van Gool. Speeded-up robust features (SURF). *CVIU*, 110(3):346–359, 2008.

[4] J. Berclaz, F. Fleuret, and P. Fua. Robust people tracking with global trajectory optimization. In *CVPR*, 2006.

[5] C. Bibby and I. Reid. Simultaneous localisation and mapping in dynamic environments (SLAMIDE) with reversible data association. In *Proceedings of Robotics Science and Systems*, 2007.

[6] C. M. Bishop. *Pattern recognition and machine learning*. Springer Verlag, 2006.

[7] E. Borenstein and S. Ullman. Class-specific, top-down segmentation. In *ECCV'02*, 2002.

[8] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *IEEE TPAMI*, 25(5):564–575, 2003.

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

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

[11] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *ECCV*, 2006.

[12] A. J. Davison. Real-time simultaneous localization and mapping with a single camera. In *ICCV*, 2003.

[13] E. Eade and T. Drummond. Scalable monocular slam. In *CVPR*, 2006.

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

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

[16] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008.

[17] 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/`.

[18] 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.

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

[20] A. Gelb. *Applied Optimal Estimation*. MIT Press, 1996.

[21] H. Grabner and H. Bischof. On-line boosting and vision. In *CVPR'06*, 2006.

[22] D. Hähnel, R. Triebel, W. Burgard, and S. Thrun. Map building with mobile robots in dynamic environments. In *ICRA'03*.

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

[24] D. Hoiem, A. A. Efros, and M. Hebert. Putting objects in perspective. In *CVPR*, 2006.

[25] M. Isard and A. Blake. CONDENSATION–conditional density propagation for visual tracking. *IJCV*, 29(1), 1998.

[26] R. Kaucic, A. G. Perera, G. Brooksby, J. Kaufhold, and A. Hoogs. A unified framework for tracking through occlusions and across sensor gaps. In *CVPR*, 2005.

[27] R. Labayrade, D. Aubert, and J.-P. Tarel. Real time obstacle detection on non flat road geometry through 'v-disparity' representation. 2002.

[28] O. Lanz. Approximate bayesian multibody tracking. *IEEE TPAMI*, 28(9):1436–1449, 2006.

[29] B. Leibe, N. Cornelis, K. Cornelis, and L. van Gool. Dynamic 3d scene analysis from a moving vehicle. In *CVPR*, 2007.

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

[31] 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.

[32] B. Leibe, K. Schindler, and L. van Gool. Coupled detection and trajectory estimation for multi-object tracking. In *ICCV*, 2007.

[33] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *CVPR*, 2005.

[34] T. Li, V. Khallem, D. Singaraju, and R. Vidal. Projective factorization of multiple rigid-body motions. In *CVPR*, 2007.

[35] A. Makadia, C. Geyer, S. Sastry, and K. Daniilidis. Radon-based structure from motion without correspondences. In *CVPR*, 2005.

[36] K. Murphy, A. Torralba, and W. T. Freeman. Using the forest to see the trees: A graphical model relating features, objects, and scenes. In *NIPS*, 2003.

[37] D. Nistér. A minimal solution to the generalised 3-point pose problem. In *CVPR*, 2004.

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

[39] K. Okuma, A. Taleghani, N. de Freitas, J. Little, and D. Lowe. A boosted particle filter: Multitarget detection and tracking. In *ECCV*, 2004.

[40] B. Ommer and J. M. Buhmann. Object categorization by compositional graphical models. In *EMMCVPR*, 2005.

[41] K. E. Ozden, K. Schindler, and L. van Gool. Simultaneous segmentation and 3d reconstruction of monocular image sequences. In *ICCV*, 2007.

[42] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann Inc., 1988.

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

[44] C. Rother, V. Kolmogorov, V. S. Lempitsky, and M. Szummer. Optimizing binary mrfs via extended roof duality. In *CVPR*, 2007.

[45] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47:7–42, 2002.

[46] K. Schindler, J. U, and H. Wang. Perspective n-view multibody structure-and-motion through model selection. In *ECCV*, 2006.

[47] S. Se, D. Lowe, and J. Little. Global localization using distinctive visual features. In *Int. Conf. Intel. Robots and Systems*, 2002.

[48] V. Sharma and J. Davis. Integrating appearance and motion cues for simultaneous detection and segmentation of pedestrians. In *ICCV*, 2007.

[49] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *CVPR*, 1999.

[50] E. B. Sudderth, A. B. Torralba, W. T. Freeman, and A. S. Willsky. Learning hierarchical models of scenes, objects, and parts. In *ICCV*, 2005.

[51] A. Torralba. Contextual priming for object detection. *IJCV*, 53(2):153–167, 2003.

[52] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: principles and practice of background maintenance. In *ICCV*, 1999.

[53] O. Tuzel, F. Porikli, and P. Meer. Human detection via classification on riemannian manifolds. In *CVPR*, 2007.

[54] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *ICCV*, 2003.

[55] C.C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte. Simultaneous localization, mapping and moving object tracking. *International Journal of Robotics Research*, 26:889–916, 2007.

[56] C. Wojek, G. Dorkó, A. Schulz, and B. Schiele. Sliding-windows for rapid object class localization: A parallel technique. In *DAGM*, 2008.

[57] 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.

[58] B. Wu and R. Nevatia. Simultaneous object detection and segmentation by boosting local shape feature based classifier. In *CVPR*, 2007.

[59] F. Yan, A. Kostin, W.J. Christmas, and J. Kittler. A novel data association algorithm for object tracking in clutter with application to tennis video analysis. In *CVPR'06*, 2006.

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

[61] Z. Zhu, T. Oskiper, O. Naroditsky, S. Samarasekera, H.S. Sawhney, and R. Kumar. An improved stereo-based visual odometry system. In *Performance Metrics for Intelligent Systems (PerMIS)*, 2006.