

Combined Image- and World-Space Tracking in Traffic Scenes

Aljoša Ošep, Wolfgang Mehner, Markus Mathias, and Bastian Leibe

Abstract—Tracking in urban street scenes plays a central role in autonomous systems such as self-driving cars. Most of the current vision-based tracking methods perform tracking in the image domain. Other approaches, *e.g.* based on LIDAR and radar, track purely in 3D. While some vision-based tracking methods invoke 3D information in parts of their pipeline, and some 3D-based methods utilize image-based information in components of their approach, we propose to use image- and world-space information jointly throughout our method. We present our tracking pipeline as a 3D extension of image-based tracking. From enhancing the detections with 3D measurements to the reported positions of every tracked object, we use world-space 3D information at every stage of processing. We accomplish this by our novel coupled 2D-3D Kalman filter, combined with a conceptually clean and extendable hypothesize-and-select framework. Our approach matches the current state-of-the-art on the official KITTI benchmark, which performs evaluation in the 2D image domain only. Further experiments show significant improvements in 3D localization precision by enabling our coupled 2D-3D tracking.

I. INTRODUCTION

Visual scene understanding in outdoor environments is a key requirement for autonomous mobile systems. The tracking and detection of traffic participants such as pedestrians, cars, and bicyclists plays an important role in safe navigation of autonomous vehicles. Through tracking, the vehicle becomes aware of the whereabouts of important objects and determines their motion.

A substantial amount of research has been done in this area, mainly driven by the goal of developing autonomous vehicles that may operate in everyday traffic. Recent advances in object detection [38], [7], [40] and detection-based multi-object tracking [9], [39], [21], [41] start to approach a matured state. However, there are still several open problems in vision-based tracking approaches. The majority of existing methods only perform tracking in the image domain. Yet, in mobile robotics and autonomous driving scenarios, precise 3D localization and trajectory estimation is of fundamental importance. In order to prevent collisions, it is crucial to be aware of the extent and the orientation of objects in world-space, especially for objects close to the camera.

In this work, we carefully combine 2D object detections and 3D stereo depth measurements in order to improve image-based tracking and, more importantly, precise 3D localization (see Fig. 1). While image-based tracking has shown to be successful even in greater distances from the camera, 3D stereo measurement precision deteriorates quickly with camera distance [29]. Our system weights these sources depending on the distance from the camera.

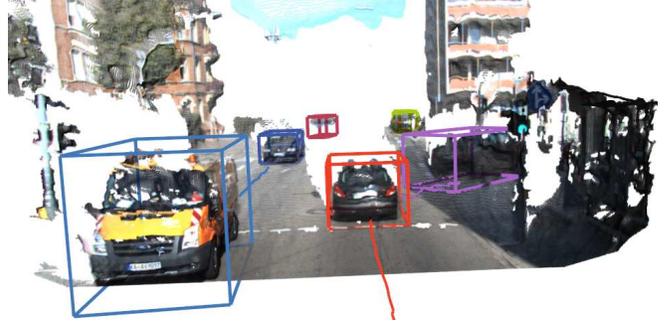


Fig. 1. Example output of our method. Using stereo matching and vision-based tracking, we obtain precise 3D bounding boxes.

It combines 2D and 3D information when available, but is also able to cope with missing 3D measurements.

Our contributions are as follows. (i) We propose a new tracking framework¹, which exploits both 2D and 3D measurements. To that end, we combine object detections (*e.g.* cars, pedestrians) and 3D object proposals obtained in a 3D point-cloud. Our method takes advantage of the strengths of both sources of information: 2D detections provide class information, while our 3D proposals assist in locating the objects in world coordinates. (ii) We introduce a novel 2D-3D Kalman filter, which is keeping both an image- and a world-space (position and size) estimate. These estimates are loosely coupled to ensure the consistency of a track. This coupling enables us to track distant objects and continue these tracks with more precise information in the close range, while smoothly transitioning between the modalities. (iii) We show competitive results on the KITTI benchmark. Additionally to these image-based evaluations, we assess the precision of our method in 3D space to quantify the advantage of our method.

II. RELATED WORK

Vision-based multi-object tracking in street scenes. Most vision-based approaches to multi-object tracking (MOT) in street scenes follow the tracking-by-detection paradigm, where object detector responses are matched across multiple frames in the image domain [9], [10], [21], [22], [30], [39], [41], [44]. Geiger *et al.* [13] associate detections using an appearance model and bounding box overlap in the image domain. Detection bounding boxes are filtered with a Kalman filter and are associated in a two stage process: first, detections are combined to short tracks (tracklets), followed by association of these tracklets to form full trajectories. Yoon *et al.* [41] propose a method that compensates for

The authors are with the Visual Computing Institute, RWTH Aachen University
E-mail: {osep, mehner, mathias, leibe}@vision.rwth-aachen.de

¹Code is available at <http://www.vision.rwth-aachen.de/page/combined-tracking>

abrupt camera motion by employing a new data association algorithm that takes structural constraints into account. Choi [9] utilizes a sparse optical flow-based descriptor as an additional affinity measure and proposes a near-online tracking formulation similar to [20]. These methods are based on monocular camera systems and perform tracking in the image domain. However, for robotics applications such as path planning and obstacle avoidance it is desirable to be fully aware of the object’s 3D position and spatial extent. Our method follows the same tracking-by-detection paradigm, but performs tracking jointly in 2D and 3D.

Vision-based multi-object tracking using depth sensors. Several vision-based tracking methods include depth information in the tracking process by using either stereo camera pairs or structured light based (RGB-D) sensors (*e.g.* Kinect, RealSense) in order to improve tracking performance. Depth information can be exploited for different purposes, *e.g.* to enhance the detection process [4], [18], [43], to reduce a detector’s search space [1], [4], [18], [24], [26], or to facilitate data association [12], [20]. While RGB-D sensors are well suited to perform pedestrian tracking in indoor scenes [4], [18], [43] they are of less use in outdoor environments due to challenging lighting conditions and reflective surfaces. Therefore, multi-object tracking systems in street scenes usually rely on a stereo-camera based setup. [12], [20], [23] use such a setup to estimate a coarse scene geometry (ground plane) and localize detections in 3D-space by ray-casting detection footpoints and intersecting them with the ground plane [20] or by performing a depth-analysis of the detection windows [12], [23]. While such a depth-analysis of detection windows has shown to work reliably for the pedestrian category, it can fail in more challenging scenarios, *e.g.* for occluded cars.

Similar to our work, [33] precisely estimates 3D pose and extent of the tracked objects. However, their method is limited to cars. We show the applicability of our approach for a multitude of object categories. For a detailed overview of MOT methods using RGB-D sensors we refer the interested reader to [6].

LIDAR-based multi-object tracking. When performing MOT using LIDAR sensors, the tracking pipeline is typically reversed using a tracking-before-detection paradigm [11], [19], [27], [34]. LIDAR sensor outputs are more precise, and do not suffer from systematic errors compared to stereo-based sensors. The acquired measurements are better suitable to delineate the shapes of objects [8]. Segmentation of object candidates from LIDAR sensor data is a well-studied problem [27], [34], [37]. These object candidates provide a precise object boundary, position and shape but no object category information, hence model-free tracking is performed (typically, using a Kalman filter and nearest-neighbor data association). Category-agnostic trajectories can then be classified into object categories [34], [37]. In our work, we also rely on category-agnostic object proposals, but rather than using expensive LIDAR systems our proposals are generated from stereo input images [28].

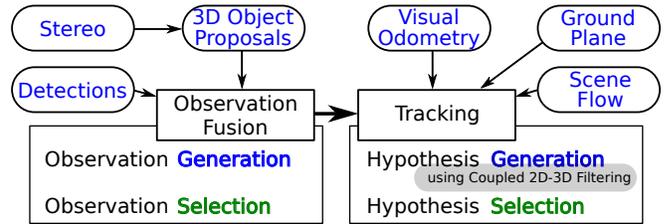


Fig. 2. Method overview. Our input data is processed in two steps. We first generate a large number of observations (observation = detection + 3D object proposal) and perform model selection to pick the most suitable ones. In the same fashion, we track the observations to generate an over-complete set of tracking hypotheses, and again perform model selection to pick the trajectories we report as results.

In the context of LIDAR tracking, it has been shown that 3D measurements of the objects can be utilized to improve 3D tracking precision (position and velocity). [17] proposes to align LIDAR object candidates in a coarse-to-fine fashion using annealed dynamic histograms in order to obtain precise position and velocity. [35] uses 3D measurements to jointly estimate trajectory and shape of the tracked object by creating an object “map”. We show that using 3D measurements is suitable for improving the 3D tracking precision even though our input data originates from noisy stereo-based depth data.

III. METHOD OVERVIEW

Fig. 2 shows an overview of our proposed pipeline. Our method combines information from several commonly used sources, such as object detections [38], [13], stereo [15], visual odometry [16], and optionally scene flow [36]. Additionally, we support the detections with stereo-based class-agnostic 3D object proposals [28], short *3D proposals*. These proposals hypothesize objects and provide precise 3D measurements for them, in our case the objects’ positions on the ground plane, physical size and a segmentation mask (See Fig. 3, bottom).

A 3D object proposal combined with a 2D detection constitutes an *observation*. We use a CRF model to select suitable observations out of the huge set of possible observations we generate. The CRF scores the observations according to the compatibility of their 2D and 3D information, and helps to exclude observations which would share either a detection or a 3D proposal with another selected observation.

Our tracker works using a comparable idea. We first generate an over-complete set of track hypotheses. They are then selected via a CRF model, which scores the hypotheses and prevents the selection of overlapping pairs of hypotheses.

In extension of the common paradigm of image-space tracking, we use 3D information in every part of the tracking pipeline. Hypotheses are tracked using Kalman filters with a joint 2D-3D state, which is weakly coupled by projection and back-projection operations. As a result, measurements of 2D bounding boxes can help to estimate the 3D position and vice versa. This also means that we can track *opportunistically*, *i.e.* we make use of the 3D measurements when available, but we can also perform the tracking without them. Finally,

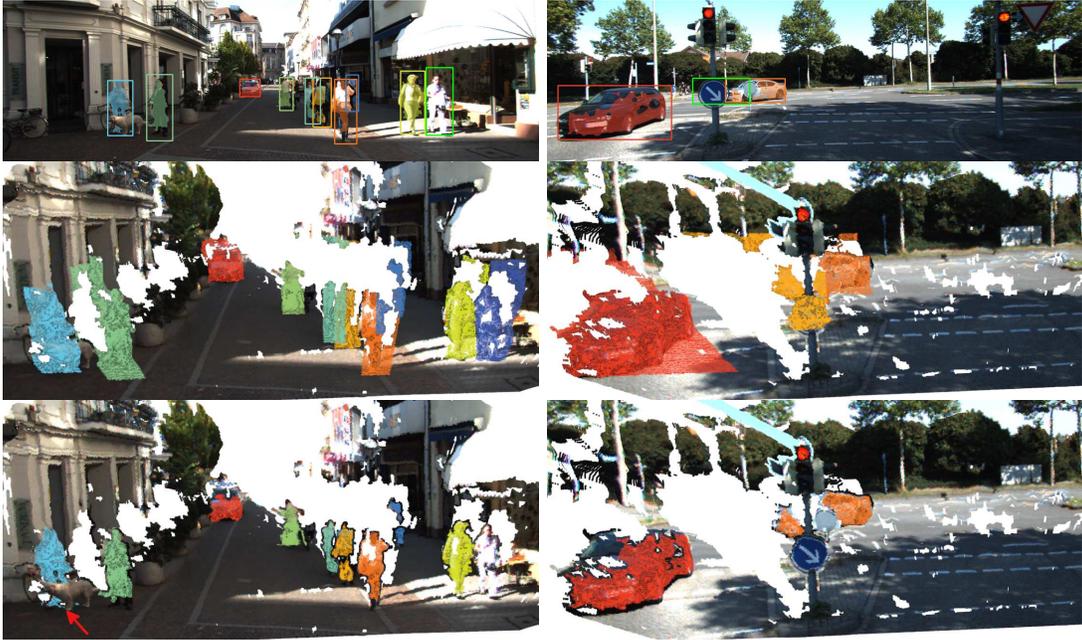


Fig. 3. Use of depth information. The top row shows detections and associated 3D object proposals (bright green boxes are not associated). The middle row shows the corresponding stereo point-clouds, where all points originating from inside the bounding boxes are highlighted. These sets of points often include an occluder or the background. The bottom row shows the same point-clouds, but points belonging to the associated 3D proposals are highlighted. The 3D proposals separate the object from the environment much better, *e.g.* the dog in the left column. *Best viewed in color.*

the CRF model scores the hypotheses by evaluating their consistency in image- and world-space.

IV. OBSERVATION FUSION MODEL

As inputs to our observation fusion we use 3D object proposals, obtained by performing clustering in the stereo point-cloud, and object detections. We fuse these sources of information before feeding them to the tracking process. This results in (i) extending the 2D detections by precise 3D measurements of the object position and size and (ii) selecting the relevant proposals from the huge set of available 3D proposals.

A. Observation Models

For 2D object detection we use image-space detectors, *e.g.* [38], [13]. A set of detections at timestep t , $t \in [0, \dots, T]$, is defined as $\mathcal{M}_{det}^t = \{m_{det}^{t,i}\}$, $i \in 1, \dots, n_t$. Each detection measurement provides the center-point, width, and height of the 2D bounding box $\mathbf{b}^{2D} = [x^{2D}, y^{2D}, w^{2D}, h^{2D}]^T$ (in the image domain), class information c , and score s_{det} :

$$m_{det}^{t,i} = [\mathbf{b}^{2D}, c, s_{det}]^T. \quad (1)$$

We obtain 3D object proposals using an extension of our previous work on multi-scale 3D proposals, described in [28]. That approach generates a large set of class-agnostic 3D object proposals by identifying clusters of depth measurements in stereo point-clouds. The intuition behind this approach is that relevant real-world objects usually stick out of the ground plane, surrounded by a certain amount of free space. As the class of each object is unknown, potential

objects have to be searched at varying scales. Clusters that remain stable over multiple scales are selected as possible object candidates. The number of proposed objects is reduced by merging clusters if their bounding boxes overlap (intersection-over-union > 0.9).

For our application, we slightly modify [28]: (i) Rather than using the projected bounding boxes in the image plane for merging the clusters, we now project 3D bounding boxes to the ground plane. This results in more precise localization in 3D. (ii) Additionally to clustering with an isotropic kernel, we add two anisotropic kernels elongated along the x- and z-direction (the two dimensions of the ground plane) in order to better represent elongated objects. As a result, we obtain a rich set of 3D object proposals $\mathcal{M}_{prop}^t = \{m_{prop}^{t,j}\}$, $j \in 1, \dots, m_t$. Each proposal is defined by its position $\mathbf{p} = [x, y, z]^T$, velocity $\mathbf{v} = [\dot{x}, \dot{y}, \dot{z}]^T$ (obtained from scene flow [36]), size estimate $\mathbf{s} = [w^{3D}, h^{3D}, l^{3D}]^T$, and score:

$$m_{prop}^{t,j} = [\mathbf{p}, \mathbf{v}, \mathbf{s}, s_{prop}]^T. \quad (2)$$

B. Observation Fusion

At this step, we have a set of detections, which provide object class information, and a set of 3D proposals, which provide localization and object size estimates. However, no category information is associated to these proposals. Therefore, a group of pedestrians is a valid object, as much as each pedestrian individually is.

Only by picking a proposal at the right scale, its measurement becomes meaningful semantically. This is a first instance of image- and world-space information supporting each other. A detection has no precise 3D information, which

is obtained by associating the detection to a proposal. On the other hand, a 3D measurement can only be considered precise (in a semantic sense) if the proposal is derived from a segment of the point-cloud which corresponds to an actual object. The set of detections helps selecting meaningful proposals, and rejecting the others. See Fig. 3 for examples of associated detections and proposals.

We address the selection problem by performing MAP inference in a CRF model. We enumerate a large set of possible associations between detections and 3D proposals. In theory, we would need to enumerate the Cartesian product of both sets to obtain the overlapping observation set $\mathbf{o}^t = \mathcal{M}_{det}^t \times \mathcal{M}_{prop}^t$. In practice we do gating and have only a limited number of overlapping observations. These associations are interpreted as nodes in a graph and we assign binary labels $\mathbf{s} \in \{0, 1\}^{|\mathbf{o}^t|}$ (1: selected, 0: not selected). By minimizing the following energy, we obtain a set of consistent associations between the proposals and the detections.

$$\mathbf{E}(\mathbf{s}, \mathbf{o}) = \sum_{o_i \in \mathbf{o}} s_i \phi(o_i) + \sum_{o_i, o_j \in \mathbf{o}} s_i s_j \varphi(o_i, o_j), \quad (3)$$

where the detection-to-proposal association potential is defined as:

$$\phi(o_i) = -w_1^o \phi_{size}(o_i) - w_2^o \phi_{pos}(o_i) - w_3^o \phi_{proj}(o_i) + w_4^o \quad (4)$$

The first term $\phi_{size}(o_i)$ scores the 3D proposal size, using the probability of the size given statistics (mean+variance) learned from data. The second term ensures that the detection and the proposal have a small distance on the ground plane (Mahalanobis distance given the uncertainty of the measurements). The third term matches the projected area of the proposal with the area of the detection bounding box (intersection-over-union). The weight w_4^o imposes a minimal requirement on the association, since observations with an overall positive score $\phi(o_i)$ will not be selected.

The pairwise term penalizes overlapping associations:

$$\varphi(o_i, o_j) = w_5^o \cdot \frac{|P_i \cap P_j|}{\min(|P_i|, |P_j|)} + w_6^o \cdot \mathbb{I}(o_i, o_j). \quad (5)$$

The first term measures the (normalized) overlap of the two observations i and j , based on the number of shared 3D points $|P_i \cap P_j|$ of their 3D proposals. The second term is a hard-exclusion term: the indicator function $\mathbb{I}(o_i, o_j)$ is 1 if two observation share a proposal or a detection and 0 otherwise. The purpose of the pairwise term is to penalize physical overlap of the observations and to disallow observations that are claiming the same proposal or detection.

The inference problem (3) is NP-hard. We therefore obtain an approximate solution using the multi-branch method from [31]. Note that the obtained solution gives us a set of valid associations between the detections and 3D proposals. As 3D proposals can mostly be obtained in the close camera range, there will be detections that are not part of any selected observation. For such detections, we augment our final observation set with *partial observations*, containing detections, but no 3D information. In particular, this helps

to retain far-away targets which are not covered by 3D proposals.

The proposed approach of combining the information from different sources is also called *early fusion*, as hard decisions are made prior to invoking the tracker. This contrasts to *late fusion*, where the tracker performs the selection and fusion of measurements. However, *late fusion* would result in a combinatorial explosion of the state-space of our multi-hypothesis tracker. Additionally, the previously described selection of observations matches the selection process of tracking hypotheses through the tracker. The scores and interaction terms consist of the same building blocks, increasing the chance that the observations picked here will produce high-scoring hypotheses.

V. TRACKING

Our tracking formulation follows a hypothesize-and-select framework, as initially proposed in [20], which is a current state-of-the-art tracking paradigm for vision-based tracking [9]. In this paradigm an over-complete set of hypotheses is created, and then the most suitable ones are selected.

In the following, we will first describe our 2D-3D Kalman filter. It is applied to filter the observations which are associated to our trajectory hypotheses. By generating an over-complete set of hypotheses we capture a multitude of possible data associations.

A. Coupled Filtering of 2D-3D States

We propose a formulation which keeps track of objects in both image domain and world-space. In contrast, most state-of-the-art vision-based tracking methods perform tracking just in the image domain.

We use an Extended Kalman Filter (EKF) to estimate a joint 2D-3D state, and couple the different quantities using projection and back-projection operations. The geometry of a hypothesis is estimated using detections (1) and 3D proposals (2), which we filter using the EKF. The state at time t is defined by $\mathbf{x}_t = [\mathbf{b}^{2D}, \dot{\mathbf{b}}^{2D}, \mathbf{p}, \mathbf{v}, \mathbf{s}]^T$. For the position \mathbf{p} on the ground plane and the 2D bounding box \mathbf{b}^{2D} we use a constant-velocity model, which requires adding the rate of change of the bounding box $\dot{\mathbf{b}}^{2D}$.

At each timestep, the bounding box position of each hypothesis is corrected for the ego-motion. The footpoint of the bounding box \mathbf{b}^{2D} is back-projected into world-space, using the estimated distance from the camera computed using \mathbf{p} . The translation and rotation given by the ego-motion is then applied to the 3D point. By projecting it back into the image, we obtain the corrected footpoint.

Then, the components of the position and bounding boxes are predicted using the corresponding velocities, except for the following heights and the footpoint estimate, which are weakly coupled through these projection and back-projection operations:

$$h^{3D} = w_b \frac{d_c}{f} h^{2D} + w_a h^{3D} \quad (6)$$

$$h^{2D} = w_a \left(\delta t \dot{h}^{2D} + h^{2D} \right) + w_b \frac{f}{d_c} h^{3D}. \quad (7)$$

$$y^{2D} = w_a (\delta t \dot{y}^{2D} + y^{2D}) + w_b \left(\frac{f}{d_c} (\delta t \dot{y}^C + y^C) + v_0 \right) \quad (8)$$

$$x^{2D} = w_a (\delta t \dot{x}^{2D} + x^{2D}) + w_b \left(\frac{f}{d_c} (\delta t \dot{x}^C + x^C) + u_0 \right) \quad (9)$$

Here δt is the time difference between two prediction steps, f is the focal length, and u_0, v_0 are the horizontal and vertical components of the principal point. The position in camera space $\mathbf{p}^C = [x^C, y^C, z^C]^T$ helps to compute the distance from camera $d_c = z^C$. The weights w_a and w_b determine how much the 2D and 3D states contribute to the coupling (we learn these weights on our training set, see Sec. VI).

Depending on which observations are available (see Sec. IV-B), we perform sequential updates with these different sources of information. We distinguish two cases: (i) Fused observation, where a detection is associated to a 3D proposal, resulting in observable values $\mathbf{z}_t = [\mathbf{b}^{2D}, \mathbf{p}, \mathbf{v}, \mathbf{s}]^T$. In this case we update the corresponding quantities of the state. The coupling only happens in the prediction step. However, a measured bounding box will still influence the next 3D prediction through the coupling. (ii) Partial observation (non-associated detections), which restricts the observable values to $\mathbf{z}_t = [\mathbf{b}^{2D}, \mathbf{p}_{bp}, \mathbf{s}_{mean}]^T$. Here, we still update the 3D position using the back-projection of the footpoint of the bounding box \mathbf{p}_{bp} , and we update the 3D size using the mean size \mathbf{s}_{mean} of the category from the training data. In this case, a different measurement variance is attached to the 3D position and size.

B. Hypothesis Set Generation

We perform tracking by maintaining an over-complete set of trajectory hypotheses $\mathbf{h} = \{\mathbf{h}_k^{t_0:t_n}\}$, each defined on the time-span $t_0:t_n$ (note that we will omit time indices where not needed). Each hypothesis is built from the set of observations using our 2D-3D Kalman filter. It is constrained to the ground plane and maintains a state estimate over time:

$$\mathbf{h}_k(t) = [\mathbf{b}^{2D}, \dot{\mathbf{b}}^{2D}, \mathbf{p}, \mathbf{v}, \mathbf{s}, \mathbf{c}, \mathbf{o}_k^t]^T. \quad (10)$$

The observation used in each frame is also attached to the hypothesis, although it may be missing for any given frame, *i.e.* $\mathbf{o}_k^t = \emptyset$. A hypothesis $\mathbf{h}_k^{t_0:t_n}$ is furthermore associated to its inlier set $\mathcal{I}_k = \{\mathbf{o}_k^t | t_0 \leq t \leq t_n\}$ of observations spread out over the temporal domain. We attach a multinomial distribution over possible object categories $\mathbf{c} = \{\text{car}, \text{pedestrian}, \text{cyclist}\}$. It is estimated from the associated detections by performing forward Bayesian filtering (likelihood terms are learned from the data).

Hypothesis Extension. We create an over-complete hypothesis set similar to the one proposed in [20]: at each time step, we (i) extend existing hypotheses with a new observation and (ii) generate alternative hypotheses starting at new observations within a temporal window.

In case several detections are very close to the hypothesis (in image-space, on the ground plane, and in appearance), we branch the hypothesis, updating each branch with a different detection.

Hypothesis Persistence. We stop updating hypotheses which have left the camera view frustum, but we keep

extrapolating them for a while (see Fig. 6). This keeps the hypothesis ‘alive’ and it continues to claim its inlier set during the optimization procedure, such that these observations are not suddenly free to support other hypotheses which would be implausible otherwise. In order to keep the size of the hypothesis set feasible, we remove duplicates and hypotheses which were not selected for some time.

C. Hypothesis Selection

In each frame we select from the set of hypotheses by performing MAP inference in a CRF model, similar to [20]:

$$\mathbf{E}(\mathbf{m}, \mathbf{h}) = \sum_{h_i \in \mathbf{h}} m_i \vartheta(h_i) + \sum_{h_i, h_j \in \mathbf{h}} m_i m_j \psi(h_i, h_j), \quad (11)$$

where $\mathbf{m} \in \{0, 1\}^{|\mathbf{h}|}$ is a binary indicator vector, with $m_i = 1$ meaning that a hypothesis has been picked. We search for the selection $\mathbf{m}^* = \operatorname{argmin}_{\mathbf{m}} \mathbf{E}(\mathbf{m}, \mathbf{h})$ with the lowest energy given a hypothesis set \mathbf{h} .

Hypothesis Score. The unary term scores a hypothesis:

$$\vartheta(h_i) = w_{min}^h - \sum_{\mathbf{o}_i^t \in \mathcal{I}_i} \mathcal{S}(\mathbf{o}_i^t, h_i), \quad (12)$$

where the model parameter w_{min}^h defines a minimal score for a hypothesis. The contribution of an observation $\mathcal{S}(\mathbf{o}_i^t, h_i^{t_0:t_n})$ is:

$$\mathcal{S}(\mathbf{o}_i^t, h_i^{t_0:t_n}) = e^{(-\tau \cdot (t_n - t))} \cdot s(\mathbf{o}_i^t) \cdot \Phi(\mathbf{o}_i^t, h_i^{t_0:t_n}). \quad (13)$$

Using an exponential decay, observations further in the past have less influence. The score of an observation is $s(\mathbf{o}_i^t) = s_{det}$. In the current formulation, we only use the score of the detection. The association affinity term:

$$\Phi(\mathbf{o}_i, h_i) = \mathbb{I}_c(\mathbf{o}_i, h_i) \cdot (w_c \Phi_c(\mathbf{o}_i, h_i) + w_m \Phi_m(\mathbf{o}_i, h_i) + w_p \Phi_p(\mathbf{o}_i, h_i)) \quad (14)$$

is a linear combination of the appearance score $\Phi_c(\mathbf{o}_i, h_i)$, the motion model term $\Phi_m(\mathbf{o}_i, h_i)$, and the projection model term $\Phi_p(\mathbf{o}_i, h_i)$, multiplied by an indicator function $\mathbb{I}_c(\cdot, \cdot)$ that prevents association between hypotheses and observations of incompatible categories. The weights are functions of the distance $d(\mathbf{o}_i)$ of an observation from the camera and the relative weighting of the appearance term w_c :

$$w_m = (1 - w_c) e^{(-\gamma \cdot d(\mathbf{o}_i))}, \quad w_p = (1 - w_c - w_m). \quad (15)$$

For the appearance score we use intersection kernels over color histograms as in [9]. The motion model term $\Phi_m(\mathbf{o}_i, h_i) \sim \mathcal{N}(\mathbf{p}_{obs} | \mathbf{p}_{pred}^{h_k}, \Sigma_{pred}^{h_k})$ scores the probability of the observation given the Kalman filter prediction (on the ground plane). The projection model $\Phi_p(\mathbf{o}_i, h_i) = \text{IoU}(\mathbf{b}^{2D}(\mathbf{o}), \mathbf{b}^{2D}(h))$ is computed as the intersection-over-union between the predicted 2D bounding box of the hypothesis and the observed bounding box (in the image domain).

Hypothesis Interaction. The pairwise potential of the CRF in (11) scores the interaction of each pair of hypotheses:

$$\psi(h_i, h_j) = w_{ol}^h O(h_i, h_j) + w_{sh}^h |\mathcal{I}_i \cap \mathcal{I}_j|. \quad (16)$$

The parameter w_{ol}^h weights the physical overlap penalty:

$$O(h_i, h_j) = \sum_t [\text{IoU}(\mathbf{b}^{2D}(h_i, t), \mathbf{b}^{2D}(h_j, t))]^2, \quad (17)$$

which punishes overlap in image space of the two hypotheses. Additionally, we add a penalty w_{sh}^h for each observation shared by the hypotheses.

Intuitively, after solving this inference problem we obtain a set of best-scoring trajectories that are physically plausible (*i.e.* do not overlap in space-time). Note the similarity of this model to the observation fusion in Sec. IV-B. The difference is that in this case we are aiming for a partition of the observations over time, whereas in Sec. IV-B we are computing associations between 3D proposals and detections.

VI. EXPERIMENTAL EVALUATION

All our evaluations are based on the KITTI Vision Benchmark Suite [14]. It provides training sequences with a publicly available ground truth, and a separate test set which can only be evaluated using the provided evaluation server. Evaluations are based on the CLEAR MOT metrics [3]. KITTI furthermore provides detections for cars, pedestrians, and bicycles from two different detectors [38], [13]. In our experiments we use the detections from the Regionlet detector [38]. We split the KITTI training data into two disjoint sets: a training set which is used for optimizing parameters (based on MOTA score) via Hyperopt [2] and a validation set for evaluating different aspects of our pipeline². We interpret the reported result mostly based on the achieved MOTA score, which is considered to be the most distinctive tracking evaluation measure [5].

Observation Precision. For our detailed analysis of the fused observations, we use 2D and 3D annotations provided in the KITTI training set. We focus on single frame results here, in order to gain insights into the observation fusion without regarding the tracking.

Fig. 4 shows the positioning error by distance range, split into the error made in estimating the distance to the camera, and the lateral error orthogonal to the depth error. We compare the performance of our 3D proposals to three baselines: (GP-P) Ray-casting of the footpoints of the 2D bounding boxes and intersection with the ground plane; (DA) Results obtained by depth analysis [12]; (3DOP) Results from the recent 3DOP [7]. While 3DOP has ultimately other goals, the estimation of 3D bounding boxes is an essential step in the pipeline.

For cars, our evaluation shows that simpler methods (GP-P and DA) are clearly outperformed by the more sophisticated ones. The lateral position of cars is best estimated by 3DOP, which is specially designed for estimating bounding boxes of specific object categories, while our method is more general. Our method is one of the two best performing methods in all cases. While DA is performing well on the *pedestrian* category, it lacks precision in the *car* category; the situation is reversed for 3DOP.

² We used sequences 1,2,5,7,8,9,11,17,18,19 as training set and sequences 0,3,4,6,10,12,13,14,15,16,20 to perform validation.

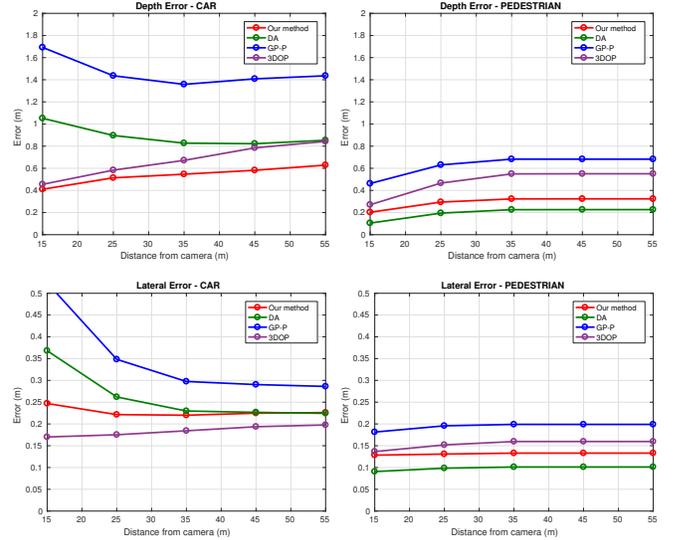


Fig. 4. Localization error by distance range in depth and lateral direction.

Ablation Study. In order to evaluate the influence of the different ingredients of our approach, we switch off parts of our pipeline. MOTA and MOTP are both computed using the 2D bounding box overlap, which is the standard in KITTI. See Table I for the results: (No-flow) uses no scene flow; (Det. only) does not use 3D proposals; (2D-tracker) is a pure 2D version of our tracking pipeline, disabling even the ground plane estimation and visual odometry; (DA) uses depth analysis [12] rather than our 3D proposals to obtain the 3D measurements. As can be observed, in this 2D evaluation our full method performs best, and each of the components contributes to the performance. A clear benefit of our method will be seen when evaluating in world space.

3D Localization Evaluation. For each true positive association we evaluate the distance in 2D (as intersection-over-union) and in 3D (as the Euclidean distance on the ground plane). This allows to compute the MOTP-2D and MOTP-3D metrics as suggested by [3]:

$$\text{MOTP-2D} = \frac{\sum_{i,t} \text{IoU}(\mathbf{b}_{gt}^i(t), \mathbf{b}_{traj}^i(t))}{\sum_t n_{tp}(t)} \quad (18)$$

$$\text{MOTP-3D} = \frac{\sum_{i,t} \|\mathbf{p}_{gt}^i(t) - \mathbf{p}_{traj}^i(t)\|}{\sum_t n_{tp}(t)}, \quad (19)$$

with the number of true positive associations per frame n_{tp} .

We compare the results using both metrics in Fig. 5. For MOTP-2D (higher values are better), there is barely any difference between the full version and the detection-only baseline. However, for MOTP-3D (lower values are better), 3D localization precision is considerably better. This experiment shows that one can benefit from using 3D information in vision-based tracking, even without compromising the ability to accurately track objects in the image domain. Given the considered applications, the MOTP-3D results are clearly more relevant, since in practical tasks we want to obtain precise information in the real world, not in image space.

Exploiting Precise 3D Segmentations. As has been shown,

TABLE I
ABLATION STUDY.

Cars	MOTA	MOTP	ID	Frag	MT	PT	ML	Pedestrians	MOTA	MOTP	ID	Frag	MT	PT	ML
Full Version	74.38	82.85	26	131	49.59	40.68	9.80	Full Version	61.87	78.85	41	164	55.95	33.33	10.71
No-flow	74.17	82.74	31	141	49.50	40.20	10.29	No-flow	61.82	78.89	53	175	54.76	34.52	10.71
Det. only	73.99	82.66	48	152	49.01	40.19	10.78	Det. only	61.13	78.88	51	172	55.95	34.52	9.52
2D-tracker	72.29	82.40	11	72	43.13	42.64	14.22	2D-tracker	59.74	78.85	59	162	48.81	35.71	15.47
DA	72.93	82.56	108	201	49.50	37.25	13.24	DA	61.69	78.97	32	148	55.95	33.33	10.71

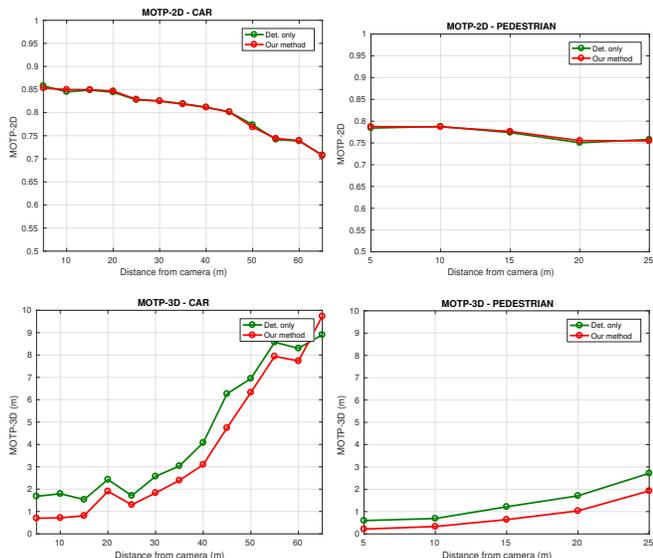


Fig. 5. Tracking Precision MOTP by distance range in 2D (higher values are better) and 3D (lower values are better) for cars and pedestrians.

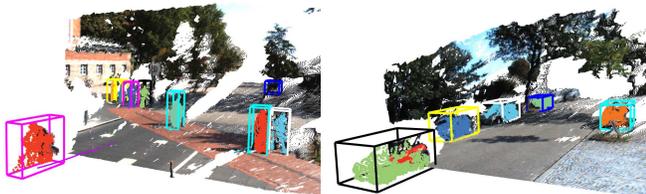


Fig. 6. Precise 3D localization enables us to do shape integration, and increase the system’s awareness of its surroundings. *Best viewed in color.*

our approach is suitable for performing precise 3D localization of tracked objects. This localization can be utilized to accumulate 3D measurements over time, in order to reconstruct more of the shape of tracked objects than can be seen in one frame. The proposals come with precise segmentations of the 3D point cloud (compare Fig. 3). The precision of these segmentation masks results in a clean shape representation. The duration of tracking helps to accumulate more depth data over time. In Fig. 6 we accumulate measurements by using the GCT representation and weighted ICP proposed by [25]. Alternatively, one could use the segments as an input to [17]. The provided result can only be obtained when performing precise segmentation in 3D. As objects and possible occluders can be well separated in world space, the shape of the tracked objects can be acquired with less noise.

KITTI Evaluation Results. Table II shows the result of



Fig. 7. Qualitative results on the KITTI tracking dataset showing 2D and 3D bounding boxes of the tracked objects.

evaluating our full pipeline on the official KITTI test set w.r.t. the highest-ranked baselines (note, that for NOMT [9], we used the online version, NOMT-HM). Although we perform tracking jointly in 2D and 3D, the official KITTI evaluation is based on bounding box overlap in the 2D image domain.

We achieve highly competitive results in both categories, cars and pedestrians. For cars, our result is on par with the best performer, NOMT-HM [9]. In contrast, for pedestrians we clearly outperform NOMT-HM. Similar to us, SCEA [41] has consistently good results on both categories and is the top performer for the pedestrians category, with our approach a close second. Fig. 7 shows a selection of qualitative tracking results.

Runtime. Our full tracking pipeline requires 347 ms per frame, excluding external components (Intel I7 CPU, single thread, not optimized). When not using observation fusion and the corresponding 3D proposals, each frame takes 48 ms.

VII. CONCLUSION

We presented a novel tracking pipeline which combines 2D and 3D measurements. Our approach shows promising results, follows a clean design and is easily extendable.

TABLE II

RESULTS ON THE KITTI BENCHMARK. TRACKING ACCURACY (MOTA) AND PRECISION (MOTP), ID-SWITCHES (ID), FRAGMENTATIONS (FRAG) MOSTLY TRACKED (MT), PARTLY TRACKED (PT), MOSTLY LOST (ML).

Cars	MOTA	MOTP	ID	Frag	MT	PT	ML	Pedestrians	MOTA	MOTP	ID	Frag	MT	PT	ML
Our method	67.35	79.25	169	675	48.93	40.09	10.98	Our method	38.37	71.44	113	912	13.40	51.55	35.05
NOMT-HM [9]	67.92	80.02	109	371	49.24	37.65	13.11	NOMT-HM	31.43	71.14	186	870	21.31	36.77	41.92
SCEA [41]	67.11	79.39	106	466	52.13	36.89	10.98	SCEA	39.34	71.86	56	649	16.15	40.55	43.30
LPSSVM [32]	66.35	77.80	63	558	55.95	35.82	8.23	LPSSVM	34.97	70.48	73	814	20.27	45.36	34.36
mbodSSP [21]	62.64	78.75	116	884	48.02	43.29	8.69	mbodSSP	-	-	-	-	-	-	-
RMOT [42]	53.03	75.42	215	742	39.48	50.46	10.06	RMOT	36.42	71.02	156	760	19.59	39.18	41.24

In future work, we plan to exploit class-agnostic tracking of objects originating from our 3D object proposals. In the context of autonomous driving cars and pedestrians are the most prominent categories, but other obstacles, from potentially unknown categories, should be identified as well.

Acknowledgements This work was funded by ERC Starting Grant project CV-SUPER (ERC-2012-StG-307432).

REFERENCES

- [1] M. Bansal, S.-H. Jung, B. Matei, J. Eledath, and H. S. Sawhney. A real-time pedestrian detection system based on structure and appearance classification. In *ICRA*, 2010.
- [2] J. Bergstra, D. Yamins, and D. D. Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *ICML*, 2013.
- [3] K. Bernardin and R. Stiefelwagen. Evaluating multiple object tracking performance: The CLEAR MOT metrics. *JVIP*, 2008:1:1–10, 2008.
- [4] D. Beymer and K. Kurt. Real-time tracking of multiple people using continuous detection. In *IEEE Frame Rate Workshop*, 1999.
- [5] S. Breuers, S. Yang, M. Mathias, and B. Leibe. Exploring bounding box context for multi-object tracker fusion. In *WACV*, 2016.
- [6] M. Camplani, A. Paiement, M. Mirmehdi, D. Damen, S. Hannuna, T. Burghardt, and L. Tao. Multiple human tracking in RGB-D data: A survey. *arXiv preprint arXiv:1606.04450*, 2016.
- [7] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun. 3D object proposals for accurate object class detection. In *NIPS*, 2015.
- [8] H. Cho, Y.-W. Seo, B. V. Kumar, and R. R. Rajkumar. A multi-sensor fusion system for moving object detection and tracking in urban driving environments. In *ICRA*, 2014.
- [9] W. Choi. Near-online multi-target tracking with aggregated local flow descriptor. In *ICCV*, 2015.
- [10] W. Choi, C. Pantofaru, and S. Savarese. A general framework for tracking multiple people from a moving camera. *PAMI*, 35(7):1577–1591, 2013.
- [11] A. Dewan, T. Caselitz, G. D. Tipaldi, and W. Burgard. Motion-based detection and tracking in 3D lidar scans. In *ICRA*, 2015.
- [12] A. Ess, B. Leibe, K. Schindler, and L. Van Gool. Robust multi-person tracking from a mobile platform. *PAMI*, 31(10):1831–1846, 2009.
- [13] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun. 3D traffic scene understanding from movable platforms. *PAMI*, 1012–1025(36):5, 2014.
- [14] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *CVPR*, 2012.
- [15] A. Geiger, M. Roser, and R. Urtasun. Efficient large-scale stereo matching. In *ACCV*, 2010.
- [16] A. Geiger, J. Ziegler, and C. Stiller. StereoScan: Dense 3D reconstruction in real-time. In *Intel. Vehicles Symp.*, 2011.
- [17] D. Held, J. Levinson, S. Thrun, and S. Savarese. Combining 3D shape, color, and motion for robust anytime tracking. In *RSS*, 2014.
- [18] O. H. Jafari, D. Mitzel, and B. Leibe. Real-time RGB-D based people detection and tracking for mobile robots and head-worn cameras. In *ICRA*, 2014.
- [19] R. Kaestner, J. Maye, Y. Pilat, and R. Siegwart. Generative object detection and tracking in 3D range data. In *ICRA*, 2012.
- [20] B. Leibe, K. Schindler, N. Cornelis, and L. V. Gool. Coupled object detection and tracking from static cameras and moving vehicles. *PAMI*, 30(10):1683–1698, 2008.
- [21] P. Lenz, A. Geiger, and R. Urtasun. FollowMe: Efficient online min-cost flow tracking with bounded memory and computation. In *ICCV*, 2015.
- [22] A. Milan, S. Roth, and K. Schindler. Continuous energy minimization for multi-target tracking. *PAMI*, 36(1):58–72, 2014.
- [23] D. Mitzel, E. Horbert, A. Ess, and B. Leibe. Multi-person tracking with sparse detection and continuous segmentation. In *ECCV*, 2010.
- [24] D. Mitzel and B. Leibe. Real-time multi-person tracking with detector assisted structure propagation. In *ICCV Workshop*, 2011.
- [25] D. Mitzel and B. Leibe. Taking mobile multi-object tracking to the next level: People, unknown objects, and carried items. In *ECCV*, 2012.
- [26] D. Mitzel, P. Sudowe, and B. Leibe. Real-time multi-person tracking with time-constrained detection. In *BMVC*, 2011.
- [27] F. Moosmann and C. Stiller. Joint self-localization and tracking of generic objects in 3D range data. In *ICRA*, 2013.
- [28] A. Osép, A. Hermans, F. Engelmann, D. Klostermann, M. Mathias, and B. Leibe. Multi-scale object candidates for generic object tracking in street scenes. In *ICRA*, 2016.
- [29] P. Pinggera, D. P. amd Uwe Franke, and R. Mester. Know your limits: Accuracy of long range stereoscopic object measurements in practice. In *ECCV*, 2014.
- [30] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes. Globally-optimal greedy algorithms for tracking a variable number of objects. In *CVPR*, 2011.
- [31] K. Schindler, J. U., and H. Wang. Perspective n-view multibody structure-and-motion through model selection. In *ECCV*, 2006.
- [32] W. Shaofei and C. C. Fowlkes. Learning optimal parameters for multi-target tracking. In *BMVC*, 2015.
- [33] S. Song and M. Chandraker. Joint SFM and detection cues for monocular 3D localization in road scenes. In *CVPR*, 2015.
- [34] A. Teichman, J. Levinson, and S. Thrun. Towards 3D object recognition via classification of arbitrary object tracks. In *ICRA*, 2011.
- [35] K. A. Ushani, N. Carlevaris-Bianco, A. G. Cunningham, E. Galceran, and R. M. Eustice. Continuous-time estimation for dynamic obstacle tracking. In *IROS*, 2015.
- [36] C. Vogel, K. Schindler, and S. Roth. Piecewise rigid scene flow. In *ICCV*, 2013.
- [37] D. Z. Wang, I. Posner, and P. Newman. What could move? Finding cars, pedestrians and bicyclists in 3D laser data. In *ICRA*, 2012.
- [38] X. Wang, M. Yang, S. Zhu, and Y. Lin. Regionlets for generic object detection. In *ICCV*, 2013.
- [39] Y. Xiang, A. Alahi, and S. Savarese. Learning to track: Online multi-object tracking by decision making. In *ICCV*, 2015.
- [40] Y. Xiang, W. Choi, Y. Lin, and S. Savarese. Data-driven 3D voxel patterns for object category recognition. In *CVPR*, 2015.
- [41] J. H. Yoon, C.-R. Lee, M.-H. Yang, and K.-J. Yoon. Online multi-object tracking via structural constraint event aggregation. In *CVPR*, 2016.
- [42] J. H. Yoon, M.-H. Yang, J. Lim, and K.-J. Yoon. Bayesian multi-object tracking using motion context from multiple objects. In *WACV*, 2015.
- [43] H. Zhang, C. Reardon, and L. E. Parker. Real-time multiple human perception with color-depth cameras on a mobile robot. *IEEE Transactions on Cybernetics*, 43(5):1429–1441, 2013.
- [44] L. Zhang, L. Yuan, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008.