

Detection and tracking of dynamic objects using 3D laser range sensor on a mobile platform

Josip Česić, Ivan Marković, Srećko Jurić-Kavelj and Ivan Petrović

University of Zagreb

Faculty of Electrical Engineering and Computing

Department of Control and Computer Engineering

Unska 3, 10000 Zagreb, Croatia

{josip.cesic, ivan.markovic, srecko.juric-kavelj, ivan.petrovic}@fer.hr

Keywords: Detection of Moving Objects, Tracking, Laser Range Sensor, JPDA Filter

Abstract: In this paper we present an algorithm for detection, extraction and tracking of moving objects using a 3D laser range sensor. First, ground extraction is performed using random sample consensus for model parameter estimation. Afterwards, to downsample the point cloud, a voxel grid filtering is executed and octree data structure is used. This data structure enables an efficient detection of differences between two consecutive point clouds, based on which clustering of dynamic parts of the cloud is performed. The obtained clusters are then expanded over the set of static voxels in order to cover entire objects. In order to account for ego-motion an iterative closest point registration technique with an initial transformation guess obtained by odometry of the platform is used. As the final step, we present a tracking algorithm based on joint probabilistic data association (JPDA) filter with variable process and measurement noise taking into account velocity and position of the tracked objects. However, JPDA filter assumes a constant and known number of objects in the scene, and therefore we use track management based on entropy. Experiments are performed using a setup consisting of a Velodyne HDL-32E mounted on top of a mobile platform in order to verify the developed algorithms.

1 INTRODUCTION

A recent development of 3D lidar technology have deepened challenges in the field of point cloud processing. Many available publications present algorithms that handle the task of detection and tracking of moving objects (DATMO), simultaneous localization and mapping (SLAM) and combination of the two (Wang, 2004). Nowadays, there exists a wide variety of approaches mostly depending on the assumed environment (e.g. indoor, outdoor, on roads, cross-country, airborne, underwater) and the expected velocity of the platform with a mounted sensor. Furthermore, ensuring real-time execution and high efficiency of such algorithms and methods, thus enabling potentially high velocities of platforms represents quite a challenging task.

In particular, dynamic scene analysis may be divided into several steps. First, whilst a point cloud is acquired, segmentation of a single scan which provides numerous objects in the scene presents a focus of the analysis. As a following task, it is necessary to extract objects with dynamic characteris-

tics, where the detection serves as an input for the tracking algorithm. Many previous works used 2D lidar technology which provides smaller point clouds containing a less eventful picture of the surrounding. Even shortly after the appearance of 3D lidar sensors, most of methods used projection of 3D point cloud onto a single plane (2D approach) or extracting a few slices from such a point cloud (2.5D approach). These approaches were widely presented on 2007 DARPA Urban Challenge (Darms et al., 2008; Montemerlo et al., 2008; Navarro-Serment et al., 2010), combined with different generative track management approaches (Petrovskaya and Thrun, 2009). So far, several works have been processing point cloud in 3D space without a projection of features so far, thus in accordance to the goal of this research, related work considered here relies on processing in 3D space. Considering different approaches, it is relevant to distinguish ones designed exclusively with an assumption on static position of the sensor (Shackleton et al., 2010; Kaestner et al., 2010; Kaestner et al., 2012), from ones that consider a moving sensor (Steinhauser et al., 2008; Moosmann et al., 2009; Moosmann and

Fraichard, 2010; Azim and Aycard, 2012), which is necessary presumption for its usage either on a vehicle or a mobile robot.

One of a pioneer works with complete solution for the DATMO problem, assuming static position of the sensor, is the work presented in (Shackleton et al., 2010). It uses a typical pipeline as a solution to the DATMO problem. By way of segmentation, classification and tracking this work has provided contribution in nullifying influence of shadows which are mostly showing up in the background of the scene. Furthermore, the solution ensures fine-grained segmentation when multiple tracked objects, in particular people, are close together. Apart from majority of works, the approach based on a stochastic Bayesian environment learning presented in (Kaestner et al., 2010) demands only a few experimentally tuned parameters. It utilizes Gaussian mixture models in order to learn 3D representations of dynamic environments where the continuous polar space around the sensor is discretized into evenly spaced range image cells in immediate encirclement of the sensor. The main objective was to determine if a measurement is caused by static or dynamic objects. This work served as a framework for classification and generative tracking approach presented in (Kaestner et al., 2012).

The work presented in (Steinhauser et al., 2008) does not provide a direct solution to the DATMO problem, it indirectly detects dynamic objects in the point cloud in order to estimate its ego motion based on provided information. In principle, it distinguishes static and dynamic objects while ego motion estimation procedure takes into account only static ones. For this purpose, after feature points and correspondences are established, random sample consensus (RANSAC) algorithm is used to classify points as static or dynamic. At the end, rotation and translation parameters are estimated. The full solution to a DATMO problem for moving platforms is presented in (Moosmann et al., 2009) and (Moosmann and Fraichard, 2010). A segmentation procedure presented in (Moosmann et al., 2009) is based on local surface geometry. Moreover, it relies upon the observation that many object parts have convex outline and that a vertical structure usually represents a single object. Given the segments within the static scene, the motion detection is achieved using a combination of local surface based feature matching and iterative closest point (ICP) algorithm (Moosmann and Fraichard, 2010), while object motion is thus estimated using Kalman filtering and dynamic mapping. Another dynamic object detection approach relies on utilization of a map of the environment (Azim and Aycard, 2012). Such an approach has high memory

and processing requirements, but has the advantage of a known environment thus avoiding segmentation of the entire static scene.

Previous discussion has brought a short overview of moving objects detection approaches, which generally serve as an input for the tracking task. Given a good detection, tracking becomes the problem of data association and state estimation. The state estimation techniques rely on methods operating within Bayesian framework, based on different aspects of Kalman filtering, grid-based approaches and approximations using particle methods (Arulampalam et al., 2002; Miller et al., 2011). Data association techniques are as well selected from a wide range of methods, some of which are optimal (e.g. multiple hypothesis tracker (MHT) (Reid, 1979)), suboptimal (e.g. probabilistic data association (PDA), joint probabilistic data association (JPDA) (Bar-Shalom, 1974)) or naive (e.g. global nearest neighbour (GNN) (Azim and Aycard, 2012)). An alternative approach to traditional multi-target tracking approaches based on a probability hypothesis density (PHD) filter where an analytical solution based on Gaussian mixtures has been presented in (Vo and Ma, 2006). This filter inherently avoids the explicit associations between measurements and targets, since it produces the mixture of probability density functions on the common state space, but it originally does not solve the problem of track extraction through time.

State-of-the-art in the field of detection can be found in review papers (Mertz et al., 2013) and (Morton et al., 2011), while detailed overview of probabilistic data association techniques is given in (Cox, 1993).

In this paper we propose a method for dynamic object detection and tracking with a 3D laser range sensor. This approach relies on the processing of two consecutive point clouds and does not use any additional sensor systems for localization apart from low cost odometry. The experiment proves the suitability of the proposed method for tasks where moderate motion of mobile platform can be assumed. We also extend the entropy based track management based on JPDA filter (Jurić-Kavelj et al., 2008) such that it takes the variable uncertainties over time into account. The uncertainties are manifested via variable process and measurement noise regarding the velocity and position of the tracked objects as well as inherent characteristics of the used sensor system.

The paper is organized as follows. The algorithm for dynamic objects extraction is given in Section 2. Section 3 presents the tracking approach based on data association using the JPDA technique, filtering based on Kalman filter and an entropy based track

management. The experimental results are presented in Section 4. Section 5 concludes the paper and provides perspectives for future work.

2 DYNAMIC OBJECT DETECTION

Static scene segmentation is a fundamental step within a dynamic object detection pipeline, where both extraction of static and dynamic objects is achieved. Afterwards, the main idea is to associate detected objects between two consecutive scans, after which objects' dynamics can be investigated. In accordance to this remark, the pipeline for dynamic objects extraction task proposed herein is mostly oriented towards exclusive detection of dynamic objects without a known map, thus influencing the complexity of track management as well as memory and processing requirements. A flow chart of the algorithm is presented in Fig. 1.

As a first step in the pipeline, ground extraction is executed. For this purpose the RANSAC method (Fischler and Bolles, 1981) was used. The goal of this robust iterative method is to estimate parameters of the plane that fits the given point cloud best. The algorithm manifests its high accuracy executed on noisy measurements common for outdoor environments as well. An eventual disadvantage of this method is its non-deterministic processing time, hence the number of iterations is limited within the application.

The ground extraction task is the only task in the pipeline which handles the entire cloud. To ensure faster execution of the algorithm, the non-ground part of the cloud is downsampled using voxel grid filtering and hierarchical octree data structure (Meagher, 1982; Wilhelms and Gelder, 2000). This filtering models occupancy of a cube with a point placed in its center if any points are located within the considered space. Octree data structure is chosen due to its proven efficiency for comparison and detection change between two consecutive point clouds (Wilhelms and Gelder, 2000).

After providing reduced, i.e. newly generated point cloud consisting of objects regardless to the ground, it is necessary to detect the dynamic ones. To extract entire objects we need to detect character of each voxel. Let S_{t-1} and S_t be the state of the voxel V in the previous scan and the current scan, respectively. Let us assume that each voxel can be modelled as *free*, *occupied* or *unobserved*. A voxel V is declared dynamic if it changes its state either from $S_{t-1} = \textit{free}$ into $S_t = \textit{occupied}$ or from $S_{t-1} =$

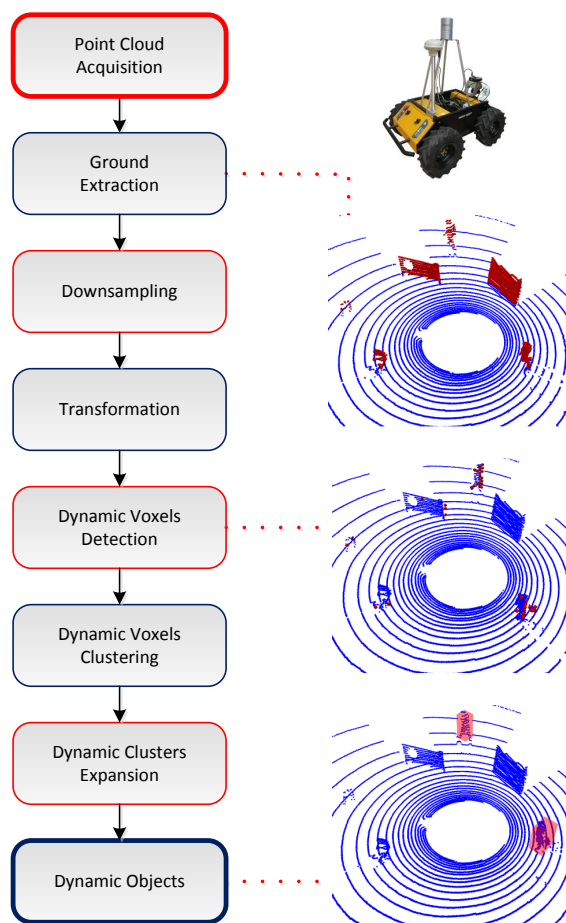


Figure 1: Flow chart of the dynamic object detection algorithm and the results of the detection

unobserved into $S_t = \textit{occupied}$. The former state change from *free* to *occupied* can clearly be declared as dynamic, while reasoning on the latter relies on the following. While moving, a mobile robot or a vehicle discovers a wide area of possibly unobserved environment in each scan. To ensure eventual security requirements and enable object detection after just a single scan, while still avoiding mapping of the environment, the considered change in the state of a voxel is rather modelled as dynamic. This can be considered as conservative approach since any new object, observed for the first time, might be detected as dynamic.

In order to enable comparison of consecutive point clouds while mobile platform is moving, transformation of clouds into a common frame represents a vital step. In contrary to some more complex but accurate localization sensing systems, such as combination of

global positioning system (GPS), wheel speed sensors and inertial measurement units (IMU) (Azim and Aycard, 2012), in this work we use well established registration technique instead, i.e. iterative closest point algorithm (ICP), originally derived in (Besl and McKay, 1992). As an optimization problem, ICP needs a variable amount of time until it converges. Thus, encoders as a single sensing system were used to get the initial guess for the registration algorithm to provide faster and more accurate convergence. It is assumed that there are enough correspondences within two consecutive scans, otherwise the convergence of the approach would be questionable. Again, due to its non-deterministic processing time, the number of iterations of ICP is bounded. An improvement on the accuracy of the transformation could have been achieved by using some more advanced estimation techniques, by extending the DATMO task with some additional applications (e.g. SLAM (Wang, 2004)) or simply by utilizing some additional sensing systems and eventually fusing the data.

Once the voxels with dynamic character are detected, a clustering over this set is executed. The clustering algorithm passes through the list of dynamic voxels defined by their center and clusters ones placed within a pre-given maximal allowed Euclidean distance with respect to considered point. Any point is allowed to be a seed point from which the cluster broadens. Previous procedure executes range search frequently, therefore an efficient kd-tree data structure is used. Since some noisy measurements appear as well, the limitation on the size of the cluster is set. Very small clusters could be caused by small dynamic objects, wrong dynamic detection of static objects or noisy measurements. All these possibilities are filtered out using such constraint, while utilization of complex algorithms to discard noisy measurements is avoided. In order to better handle big or slow objects whose shift between two consecutive scans does not go beyond the size of the object, since the map of the environment is not used, the need for broadening the obtained dynamic clusters with static occupied voxels appears. After the expansion step, some simple heuristics is applied in order to discard detections that could unlikely correspond to objects of interest (e.g. for an outdoor scene, if a height of detected cluster goes beyond some value, it likely does not correspond to neither person, nor cyclist). Examples of the detection procedure for an indoor and an outdoor scene are shown in Fig. 2 and Fig. 3, respectively.

The left figure in Fig. 2 shows the point cloud gathered at the faculty lobby with multiple moving persons. Figure in the middle shows the result of the ground extraction (blue) and dynamic clusters after

expansion step (red). Right figure presents the result of dynamic objects detection after applying threshold conditions on dynamic clusters, where multiple correct detections as well as one false alarm placed on top of the figure have appeared.

The left figure in Fig. 3 shows the point cloud gathered at the faculty courtyard with multiple moving objects. Figure in the middle shows the result of the ground extraction (blue) and dynamic clusters after expansion step (red). Right figure presents the result of dynamic objects detection after applying threshold conditions on dynamic clusters, where two cyclists and one moving pedestrian have appeared.

Since no map of the environment is built, some false alarms might appear as a drawback, especially if the sensor is placed on a fast platform or a car. On the other hand, the memory and processing requirements are kept quite low. Still, even some false positives appear, they can be filtered out using convenient data association approach. Therefore, a generic tracking management is presented in the following section.

3 DYNAMIC OBJECT TRACKING

After providing positions of the objects using previously presented algorithm, tracking task stands as the second major part within the DATMO problem solution. Herein the tracking task is divided into three parts: data association, filtering and generative track management. As stated in the introductory section, we use the JPDA filter with Kalman filtering and entropy based track management.

3.1 Data association

Consider the following sets

$$\begin{aligned} \mathbf{X}^k &= \{\hat{x}_1^k, \hat{x}_2^k, \dots, \hat{x}_{T_k}^k\}, \\ \mathbf{Z}_k &= \{z_1^k, z_2^k, \dots, z_{m_k}^k\}, \\ \mathbf{Z}^k &= \{\mathbf{Z}_1, \mathbf{Z}_2, \dots, \mathbf{Z}_k\}, \end{aligned} \quad (1)$$

where set \mathbf{X}^k is a set of continuous random variables that represent initialized tracks at time k with T_k denoting the number of tracks. The set \mathbf{Z}_k represents measurements, i.e. detections of dynamic objects at time k with m_k being a total number of detected objects within a particular scan. The last given set \mathbf{Z}^k contains all measurements received until and including moment k .

Let Θ^{k-1} denote a set of hypotheses about measurement to track assignment at time $k-1$ and $\Theta_{p(h)}^{k-1}$ denote a specific hypothesis at the time. Considering a specific hypothesis and set of measurements

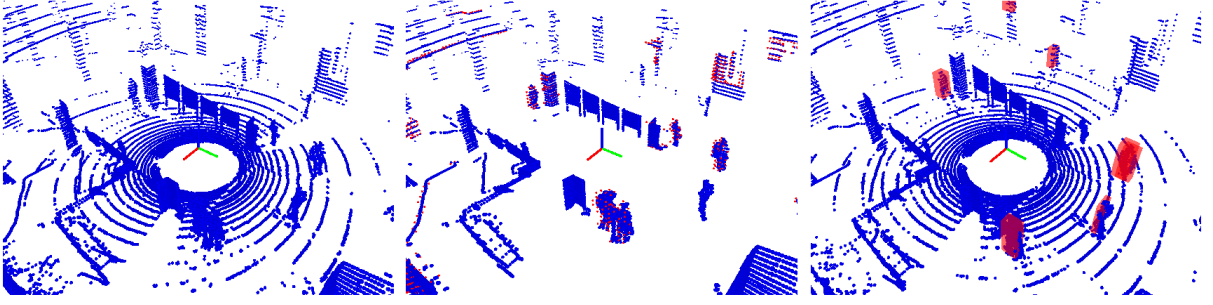


Figure 2: The result of dynamic objects detection procedure for an indoor scene. The left figure shows the point cloud gathered at the faculty lobby with multiple moving persons. Figure in the middle shows the result of the ground extraction (blue) and dynamic clusters after expansion step (red). Right figure presents the result of dynamic objects detection after applying threshold conditions on dynamic clusters, where multiple correct detections as well as one false alarm placed in top of the figure have appeared. The red markers representing the objects are put over the original point cloud for the clarity.

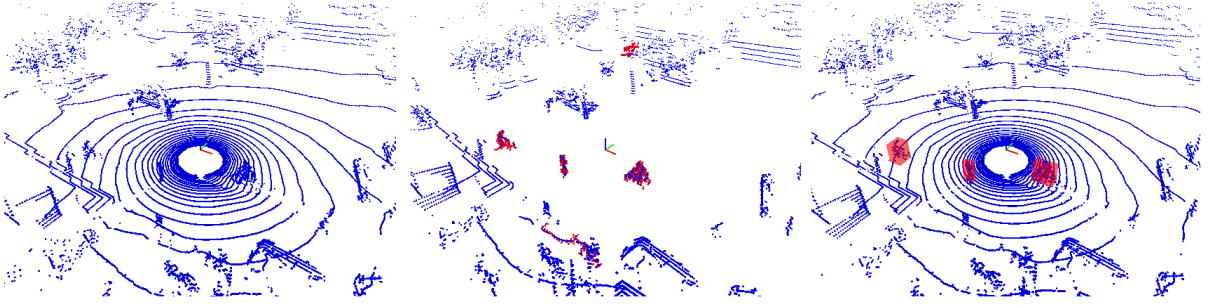


Figure 3: The result of dynamic objects detection procedure for an outdoor scene. The left figure shows the point cloud gathered at the faculty courtyard with multiple moving objects. Figure in the middle shows the result of the ground extraction (blue) and dynamic clusters after expansion step (red). Right figure presents the result of dynamic objects detection after applying threshold conditions on dynamic clusters, where two cyclists and one moving pedestrian have appeared. The red markers representing the objects are put over the original point cloud for the clarity.

\mathbf{Z}_k , we can construct legal assignment $\theta_h(k)$. The resulting hypothesis at time k is denoted as $\Theta_h^k = \{\Theta_{p(h)}^{k-1}, \theta_h(k)\}$. The JPDA filter can be viewed as a special case of MHT where in each iteration all considered hypotheses are reduced to a single hypothesis $\theta(k)$. Thus, a probability for such an assignment determined by particular hypothesis reduces to

$$P(\Theta_h^k | \mathbf{Z}^k) = \frac{1}{c} \cdot P(\mathbf{Z}_k | \theta_h(k), \theta(k-1), \mathbf{Z}^{k-1}) \cdot P(\theta_h(k) | \theta(k-1), \mathbf{Z}^{k-1}) \cdot P(\theta(k-1) | \mathbf{Z}^{k-1}), \quad (2)$$

where c is a normalizing constant. The probability of the last term in (2), i.e. $P(\theta(k-1) | \mathbf{Z}^{k-1})$, equals to unity since $\theta(k-1)$ is the only hypothesis left after the measurements processing. The second term $P(\theta_h(k) | \theta(k-1), \mathbf{Z}^{k-1})$ can be modelled as a constant (Schulz et al., 2003). Therefore, we need to develop

only the first term

$$P(\mathbf{Z}_k | \theta_h(k), \theta(k-1), \mathbf{Z}^{k-1}) = P(\mathbf{Z}_k | \theta_h(k)) = \prod_{j=1}^{m_k} P(z_j^k | \theta_h(k)), \quad (3)$$

where $P(z_j^k | \theta_h(k))$ depends on the measurement-to-track association made by hypothesis $\theta_h(k)$

$$P(z_j^k | \theta_h(k)) = \begin{cases} P_F, & z_j^k \text{ false alarm} \\ P_D P(z_j^k | \hat{\mathbf{x}}_i^k), & z_j^k \text{ existing track} \end{cases} \quad (4)$$

where P_F and P_D are the probabilities of a false alarm and detection, respectively.

Due to the presented character of JPDA filter, namely the fact that hypotheses tree is reduced to a single branch $\theta(k)$, the association probabilities are given as follows

$$\beta_j^t = \sum_{\theta \in \Theta_{jt}^k} P(\theta | \mathbf{Z}^k), \quad (5)$$

where Θ_{jt}^k denotes all the hypotheses that associate measurement j with track t at time k .

By inserting (3) into (2) and then inserting result into (5), we get an expression for measurement-to-track association probabilities

$$\beta_j^t = \frac{1}{c} \sum_{\theta \in \Theta_{jt}^k} \prod_{j=1}^{m_k} P(z_j | \theta). \quad (6)$$

These association probabilities are used to update the track states with all the measurements from the cluster, while after the update only one hypothesis remains - the one with the current track states.

3.2 Kalman JPDA filter

Herein we use a quite general constant velocity model for motion estimation in horizontal plane. State is described by position (x, y) and velocity (\dot{x}, \dot{y}) in 2D, as $\mathbf{x} = [x \ \dot{x} \ y \ \dot{y}]^T$. The model itself is given by

$$\begin{aligned} \mathbf{x}^{k+1} &= \mathbf{F}\mathbf{x}^k + \mathbf{G}w_k \\ &= \begin{bmatrix} 1 & \Delta T_k & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta T_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}^k + \begin{bmatrix} \frac{\Delta T_k^2}{2} & 0 \\ \Delta T_k & 0 \\ 0 & \frac{\Delta T_k^2}{2} \\ 0 & \Delta T_k \end{bmatrix} w_k, \end{aligned} \quad (7)$$

where w_k is the process noise and ΔT_k is the update interval. Prediction is calculated using standard Kalman filter equations

$$\begin{aligned} \hat{\mathbf{x}}_t^{k-} &= \mathbf{F}\hat{\mathbf{x}}_t^{k-1}, \\ \mathbf{P}_t^{k-} &= \mathbf{F}\mathbf{P}_t^{k-1}\mathbf{F}^T + \mathbf{G}\mathbf{Q}_t^k\mathbf{G}^T, \end{aligned} \quad (8)$$

where \mathbf{Q}_t^k is the process noise covariance matrix. Due to the inherent character of the sensing system, the acquisition of a point cloud can not be executed instantly. Therefore, while tracked object is being scanned (at 10 Hz rate), it moves as well (more technical details follow in the next section). This effect is especially noticeable while fast objects pass close to the lidar, placed on a mobile platform, and causes errors in the position of detected object. One way to handle the problem would be taking it into account within the detection procedure, but it might require complex transformations of a point cloud. On the other hand, it can be built-in in the process noise weighted in the direction of the object's movement. In the model given by (7), noise is modelled as acceleration of the track. A convenient way to model maximal acceleration a_{max} is given with $a_{max} = \frac{v_{max}^2}{r_{min}}$, where v_{max} stands for a maximal anticipated velocity and r_{min} stands for minimal distance between the object and

the lidar. The previous observation can be considered as process noise included into acceleration weighted in the predicted direction proportionally to the predicted velocity.

The innovation vector v_j^t and its covariance are \mathbf{S}_t

$$\begin{aligned} v_j^t &= \mathbf{x}_j - \mathbf{H}\hat{\mathbf{x}}_t^{k-}, \\ \mathbf{S}_t &= \mathbf{H}\mathbf{P}_t^{k-}\mathbf{H}^T + \mathbf{R}, \end{aligned} \quad (9)$$

where \mathbf{H} is the output matrix and \mathbf{R} is the measurement noise covariance matrix. Since majority of lidar systems scan the environment in a way that laser rays drift radially from each other, objects scanned further from it have sparser point cloud representation. Due to this effect, in accordance to the expected distance of the observed object, it is necessary to adapt maximal allowed distance between two neighbouring points to be clustered. Thus, the uncertainty of the position of detected object grows with its distance. Therefore, we model linear relation between measurement standard deviation and predicted distance of the observed object.

In order to lower the processing requirements we have utilized a measurement gating. In particular, since the innovation term $v_j^t \mathbf{S}_t^{-1} v_j^t$ has χ^2 distribution by using tables we can select upper limit which includes valid measurements with, e.g. 99% probability.

The update step is done using weighted innovation v_t and standard Kalman gain \mathbf{K}_k as

$$\begin{aligned} v_t &= \sum_{j=1}^{m_k} \beta_j^t v_j^t, \\ \hat{\mathbf{x}}_t^k &= \hat{\mathbf{x}}_t^{k-} + \mathbf{K}_k v_t \\ \mathbf{K}_k &= \mathbf{P}_t^{k-} \mathbf{H}^T \mathbf{S}_t^{-1}. \end{aligned} \quad (10)$$

while the covariance update is calculated as in (Blackman and Popoli, 1999)

$$\mathbf{P}_t^k = \beta^t \mathbf{P}_t^{k-} + (1 - \beta^t) [\mathbf{I} - \mathbf{P}_k \mathbf{H}] \mathbf{P}_t^{k-} + \mathbf{K}_k \mathbf{P}_{v^t} \mathbf{K}_k^T, \quad (11)$$

where

$$\begin{aligned} \beta_t &= 1 - \sum_{j=1}^{m_k} \beta_j^t, \\ \mathbf{P}_{v^t} &= \sum_{j=1}^{m_k} \beta_j^t v_j^t (v_j^t)^T - v^t (v^t)^T. \end{aligned} \quad (12)$$

3.3 Track management

Due to an inherent characteristic of JPDA filter which assumes known and constant number of tracked objects, it is necessary to design generative track management which handles the number of objects in the

scene. A solution for Kalman filter, described in (Blackman and Popoli, 1999), is based on logarithmic hypothesis ratio and innovation matrix. Another approach, presented in (Schulz et al., 2003), proposes usage of a Bayesian estimator of the number of objects for an LRS.

In the work (Jurić-Kavelj et al., 2011), an approach based on entropy measure as a feature in track management was used. It gives a basis for track management that can be readily utilized independently of the filtering approach, where all the information required for the entropy calculation are available in the running filter and the sensor model. A practical measure for this task is the quadratic Rényi entropy (Rényi, 2007)

$$H_2(\hat{\mathbf{x}}_t) = -\log \int p(\hat{\mathbf{x}}_t)^2 d\hat{\mathbf{x}}_t. \quad (13)$$

In the case of Gaussian distribution an analytical solution is given by

$$H_2(\hat{\mathbf{x}}_t) = \frac{n}{2} \log 4\pi + \frac{1}{2} \log |\mathbf{P}_t|, \quad (14)$$

where n is the state dimension, \mathbf{P}_t is the covariance matrix ($|\mathbf{P}_t| = \prod_{i=1}^n \lambda_i$, where λ_i is \mathbf{P}_t 's i -th eigenvalue). Although the Shannon entropy can also be calculated in closed form for the Gaussian distribution, the Rényi entropy was chosen in (Jurić-Kavelj et al., 2011) since it enabled closed form for the case of the particle filter approximated with a mixture of Gaussian distributions.

As discussed in previous subsection, the process noise depends on the velocity of the moving object, having the highest value in the direction of movement. Since this observation influences entropy, it is needed to tolerate higher uncertainty in the direction of movement to keep a track alive. For this purpose, we propose an approach which includes a modification of eigenvalues for the calculation of entropy. The modified eigenvalues $\lambda_{i,mod}$ are calculated as follows

$$\lambda_{i,mod} = \lambda_i \left[\alpha + (1 - \alpha) \frac{1 - \|\text{Proj}(\mathbf{l}_{i,v}, \mathbf{v})\| \frac{\|\mathbf{v}\|}{v_{max}}}{\|\mathbf{l}_{i,v}\|} \right], \quad (15)$$

where \mathbf{v} is the estimated velocity ($\mathbf{v} = \hat{\mathbf{x}}_t \circ [0101]$; \circ stands for element-wise product), $\mathbf{l}_{i,v}$ is a vector consisting of components of i -th eigenvector \mathbf{l}_i of \mathbf{P}_t related to velocity ($\mathbf{l}_{i,v} = \mathbf{l}_i \circ [0101]$) and $\alpha \in [0,1]$ is a constant. $\text{Proj}(\mathbf{a}, \mathbf{b})$ is projection of vector \mathbf{a} onto \mathbf{b} . A term for modified entropy calculation is

$$H_{2,mod}(\hat{\mathbf{x}}_t) = \frac{n}{2} \log 4\pi + \frac{1}{2} \log \prod_{i=1}^n \lambda_{i,mod}. \quad (16)$$

From (14), (15) and (16), as long as $\|\mathbf{v}\| < v_{max}$, it follows that $H_{2,mod} < H_2$. By (15), we lower i -th eigenvalue which corresponds to i -th eigenvector

proportionally with the length of projection of i -th eigenvector onto velocity of the tracked object. This way we want to tolerate uncertainty in the movement direction. This reasoning follows from the discussion related to the process noise, where even aware of higher uncertainty in track state we still want to keep track alive. In contrary, some faster dynamic objects might lose their track.

The threshold setting suits the track management logic as follows. When the track is initialized it is considered as tentative and the initial entropy is stored. When the entropy of the tentative track drops under confirmation threshold, considered track is confirmed. Afterwards, once the entropy gets over deletion threshold, the track is deleted. Furthermore, no entropy should be greater than the one calculated at the point of the track initialization.

4 EXPERIMENTAL RESULTS

To evaluate the proposed algorithms an experimental setup consisting of a Velodyne HDL-32E High Definition Lidar mounted on top of a Husky A200 mobile platform was used. The lidar sensor used herein has 32 lasers across 40° vertical and 360° horizontal field of view. The lasers are aligned vertically from -30° to 10°. It generates approximately 700,000 points per second with a range of 70 m. It rotates at a rate of 10 Hz thus producing approximately 70,000 points per turn.

In order to show the efficiency and usefulness of the proposed methods, the experiments were carried out in two different scenarios. First experiment was performed in a complex indoor environment, whilst another experiment was carried out in a complex outdoor scene.

Indoor scene. The experiment was performed having several persons walking in the surrounding of a mobile robot. The results of the detection task for this scenario are presented in Fig. 2. The results of the tracking task for the outdoor scene are shown in Fig. 4. Left figure shows the trajectories and starting points of the moving objects (two green lines, two magenta lines and two blue lines correspond to six persons starting at the positions of squares, while black line corresponds to a wrong track confirmation), the mobile platform (red line starting at the position of the circle) and several tentative filters caused by false alarms (pluses). Upper-right and bottom-right figures respectively show velocities and entropies of the moving objects, the mobile platform as well as several tentative filters (pluses) that were not confirmed as

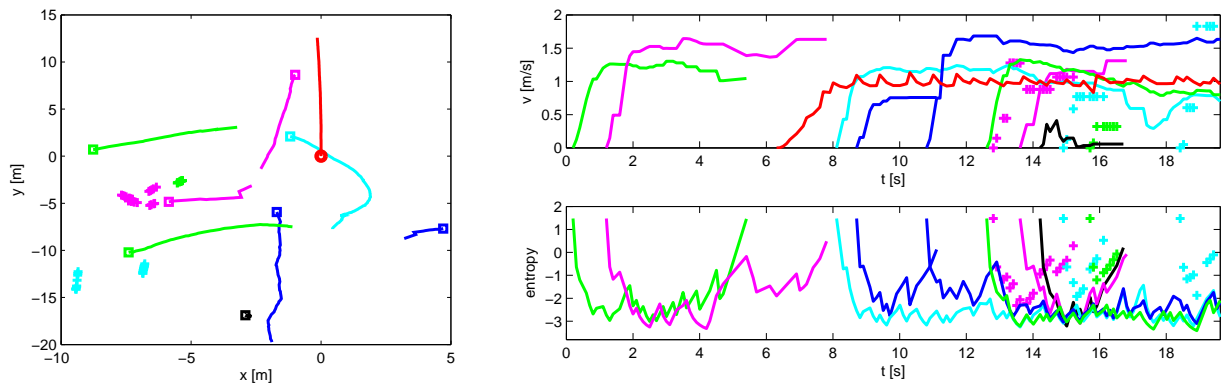


Figure 4: The experimental results of DATMO procedure for a complex indoor scene. Left figure shows the trajectories and starting points of the moving objects (two green lines, two magenta lines and two blue lines correspond to six persons starting at the positions of squares, while black line corresponds to a wrong track confirmation), the mobile platform (red line starting at the position of the circle) and several tentative filters caused by false alarms (pluses). Upper-right and bottom-right figures respectively show velocities and entropies of the moving objects, the mobile platform as well as several tentative filters (pluses) that were not confirmed as tracks.

tracks.

Outdoor scene. Alongside with several walking persons, two cyclists have also appeared in the second experiment. The results of the detection task for this scenario are presented in Fig. 3. The results of the tracking task for the outdoor scene are shown in Fig. 5. Left figure shows the trajectories and starting points of the moving objects (magenta and green line correspond to two cyclists, while black, yellow and blue present three pedestrians starting at the positions of squares) and the mobile platform (red line starting at the position of the circle) and several tentative filters caused by false alarms (pluses). Upper-right and bottom-right figures respectively show velocities and entropies of the moving objects, the mobile platform as well as several tentative filters (pluses) that were not confirmed as tracks. Due to physical constraint of the lidar, the magenta cyclist was not detected in the proximity of the sensor, hence its track was divided into two segments.

In this experiment, with approximately 200 processed clouds for both scenarios, several walking pedestrians and two cyclists were correctly tracked although many objects (both static and dynamic) were present. Pluses on both Figs. 4 and 5 correspond to tentative filters caused by false alarms appearing mostly due to strong vibrations of the platform, but only one such track was confirmed. It is also suitable to note that none of the dynamic objects has passed obscured. Since we had none database with known ground truth available, the comparison of the ground truth data with the results of our DATMO approach has not been provided. Nevertheless, presenting these two experiments, the method is clearly suitable for

performing detection and tracking task efficiently and reliably when moderate motion of a mobile platform can be assumed.

The experiments were performed on a machine running at 2.4 GHz and the algorithm was executing at 5 Hz. The data were collected and processed using the Robot Operating System (ROS) (Quigley et al., 2009) and Point Cloud Library (PCL) (Rusu, 2014).

5 CONCLUSION

In this paper we have addressed the DATMO problem using a 3D laser range sensor on a vehicle. The proposed detection pipeline consists of ground extraction, downsampling of the point cloud and the detection of dynamic parts of space, namely voxels. The dynamic voxels detection is executed by comparison of two consecutive point clouds based on the ICP algorithm with an initial transformation guess obtained by odometry, after which the clustering was performed. The tracking task used JPDA filter and Kalman filtering. The algorithm also uses the modified track management to enable variable number of tracked objects. Within proposed tracking approach an adaptive process and measurement noise, that inherently take into account characteristics of used sensor as well as track state, are modelled. The results have conformed that the presented algorithms can successfully perform the detection and tracking of moving objects.

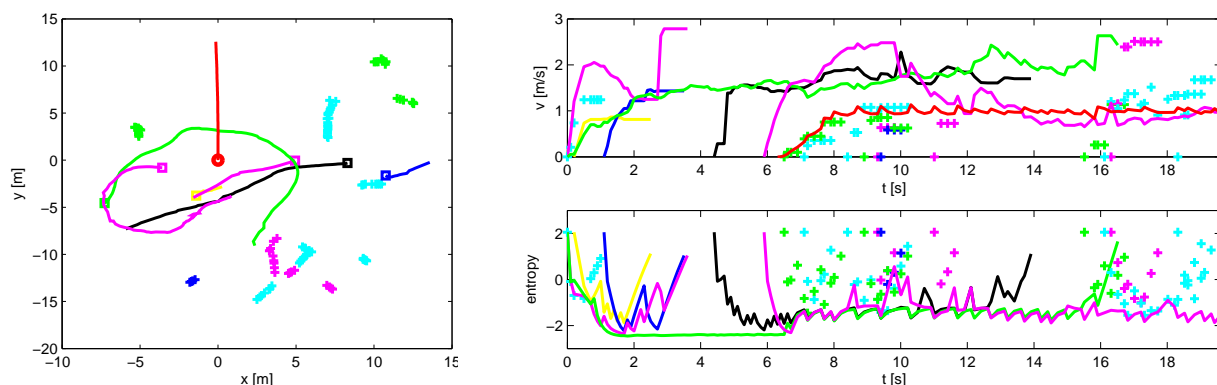


Figure 5: The experimental results of DATMO procedure for a complex outdoor scene. Left figure shows the trajectories and starting points of the moving objects (magenta and green line correspond to two cyclists, while black, yellow and blue present three pedestrians starting at the positions of squares) and the mobile platform (red line starting at the position of the circle) and several tentative filters caused by false alarms (pluses). Upper-right and bottom-right figures respectively show velocities and entropies of the moving objects, the mobile platform as well as several tentative filters (pluses) that were not confirmed as tracks.

ACKNOWLEDGEMENTS

This work has been supported by research project VISTA (EuropeAid/131920/M/ACT/HR) and European Community's Seventh Framework Programme under grant agreement no. 285939 (ACROSS).

REFERENCES

- Arulampalam, M., Maskell, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for on-line nonlinear/non-gaussian bayesian tracking. *Signal Processing, IEEE Transactions on*, 50(2):174–188.
- Azim, A. and Aycard, O. (2012). Detection, classification and tracking of moving objects in a 3d environment. In *Intelligent Vehicles Symposium*, pages 802–807. IEEE.
- Bar-Shalom, Y. (1974). Extension of the probabilistic data association filter to multitarget environment. *Proc. Fifth Symp. on Nonlinear Estimation*.
- Besl, P. J. and McKay, N. D. (1992). A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256.
- Blackman, S. and Popoli, R. (1999). *Design and Analysis of Modern Tracking Systems*. Artech House Radar Library. Artech House.
- Cox, I. J. (1993). A review of statistical data association techniques for motion correspondence. *International Journal of Computer Vision*, 10:53–66.
- Darms, M., Rybski, P., and Urmson, C. (2008). Classification and tracking of dynamic objects with multiple sensors for autonomous driving in urban environments. In *Intelligent Vehicles Symposium, 2008 IEEE*, pages 1197–1202.
- Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395.
- Jurić-Kavelj, S., Đakulović, M., and Petrović, I. (2008). Tracking multiple moving objects using adaptive sample-based joint probabilistic data association filter. In *Proceedings of 5th International Conference on Computational Intelligence, Robotics and Autonomous Systems (CIRAS 2008)*, pages 93–98.
- Jurić-Kavelj, S., Marković, I., and Petrović, I. (2011). People tracking with heterogeneous sensors using jpdaf with entropy based track management. In *Proceedings of the 5th European Conference on Mobile Robots (ECMR2011)*, pages 31–36.
- Kaestner, R., Engelhard, N., Triebel, R., and Siegwart, R. (2010). A bayesian approach to learning 3d representations of dynamic environments. In *Proceedings of The 12th International Symposium on Experimental Robotics (ISER)*, Berlin. Springer Press.
- Kaestner, R., Maye, J., and Siegwart, R. (2012). Generative object detection and tracking in 3d range data. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*.
- Meagher, D. (1982). Geometric modeling using octree encoding. *Computer Graphics and Image Processing*, 19(2):129–147.
- Mertz, C., Navarro-Serment, L. E., and MacLachlan (2013). Moving object detection with laser scanners. *Journal of Field Robotics*, 30(1):17–43.
- Miller, I., Campbell, M., and Huttenlocher, D. (2011). Efficient unbiased tracking of multiple dynamic obstacles under large viewpoint changes. *Trans. Rob.*, 27(1):29–46.
- Montemerlo, M., Becker, J., Bhat, S., and Dahlkamp, H. (2008). Junior: The stanford entry in the urban challenge. *J. Field Robot.*, 25(9):569–597.

- Moosmann, F. and Fraichard, T. (2010). Motion estimation from range images in dynamic outdoor scenes. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 142–147.
- Moosmann, F., Pink, O., and Stiller, C. (2009). Segmentation of 3d lidar data in non-flat urban environments using a local convexity criterion. In *Intelligent Vehicles Symposium, 2009 IEEE*, pages 215–220.
- Morton, P., Douillard, B., and Underwood, J. (2011). An evaluation of dynamic object tracking with 3d lidar. In *2011 Australasian Conference on Robotics and Automation (ACRA)*. ACRA.
- Navarro-Serment, L. E., Mertz, C., and Hebert, M. (2010). Pedestrian detection and tracking using three-dimensional lidar data. *The International Journal of Robotics Research, Special Issue on the Seventh International Conference on Field and Service Robots*, 29(12):1516 – 1528.
- Petrovskaya, A. and Thrun, S. (2009). Model based vehicle detection and tracking for autonomous urban driving. *Auton. Robots*, 26(2-3):123–139.
- Quigley, M., Gerkey, B., Conley, K., Faust, J., Foote, T., Leibs, J., Berger, E., Wheeler, R., and Ng, A. (2009). ROS: an open-source Robot Operating System. *IEEE Int. Conf. on Robotics and Automation (ICRA), Workshop on Open Source*.
- Reid, D. (1979). An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854.
- Rényi, A. (2007). *Probability Theory*. Dover books on mathematics. Dover Publications, Incorporated.
- Rusu, R. B. (2014). *The Point Cloud Library (PCL)*.
- Schulz, D., Burgard, W., Fox, D., and Cremers, A. B. (2003). People tracking with mobile robots using sample-based joint probabilistic data association filters. *The International Journal of Robotics Research*, 22(2):99–116.
- Shackleton, J., VanVoorst, B., and Hesch, J. (2010). Tracking people with a 360-degree lidar. In *Proceedings of the 2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS '10*, pages 420–426, Washington, DC, USA. IEEE Computer Society.
- Steinhauser, D., Ruepp, O., and Burschka, D. (2008). Motion segmentation and scene classification from 3d lidar data. In *Intelligent Vehicles Symposium, 2008 IEEE*, pages 398–403.
- Vo, B.-N. and Ma, W.-K. (2006). The Gaussian mixture probability hypothesis density filter. *IEEE Transactions on Signal Processing*, 54(11):4091–4104.
- Wang, C.-C. (2004). *Simultaneous Localization, Mapping and Moving Object Tracking*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.
- Wilhelms, J. and Gelder, A. V. (2000). Octrees for faster isosurface generation. *IEEE Transactions on Medical Imaging*, 19:739–758.