Detection and Tracking of Moving Objects

Stiven S. Dias
Instituto Tecnológico de Aeronáutica
stivendias@gmail.com

Abstract

This document presents a preliminary design architecture of a CARMEN module for Detection and Tracking of Moving Objects (DATMO).

1 Introduction

We present in this document a preliminary conceptual architecture of a new CARMEN module for Detection and Tracking of Moving Objects (DATMO).

Classical approach The approach employed by the majority of state of the art tracking systems to solve the multiple target tracking (MTT) problem is to break it into several single target tracking (STT) problems. Roughly speaking, the generated measurements at each sensor scan are first associated with existing tracks[1] maintained by the MTT system to represent the targets in the observed scenario. In general, the data association algorithms assume that, at each time scan, there is at most one valid measurement for each target being observed. For example, we usually assume that a radar can detect a point target only once per sector scan. The generated measurement-to-track association pairs are then employed to individually update the estimated target state. More precisely, each track assimilates the corresponding measurement by means of some implementation – e.g. Kalman filter (KF), extended Kalman filter (EKF), particle filter (PF) – of the standard Bayesian filter. Measurements not associated with any track are employed to create and initialize new tentative tracks. Moreover, tracks are confirmed or deleted according to the amount of evidence of its existence in the real world accumulated by the tracker along multiple sensor scans. Note that it is left to the tracker to definitely decide, over multiple frames (i.e. sensor scans), if a measurement is a false alarm or it truly represents a target in the scenario. However, usually, the detection physical phenomenon is not explicitly modeled and the targets are assumed to be always present in the observed scenario. Please refer to [1] for a concisely introductory review on the design of classic MTT systems.

In our particular DATMO problem, the vehicles are best modeled as extended objects such as tri-dimensional (3D) boxes. Furthermore, the physical sensor, i.e. the Velodyne, produces at each scan multiple depth measurements for each observed vehicle. Thus, to employ the classical approach in our particular DATMO problem, we should be able to produce higher level measurements that could individually originate at a single target in the observed scenario. Ideally, the higher level virtual sensor should detect 3D boxes by processing the low level depth measurements. On the other hand, since the box-shaped vehicles are only partially observed by the Velodyne sensor mounted on the top of the robot, a 3D box detector will probably require multiple frames to detect/identify a 3D box in the scene. Thus, besides being significantly complex and processing demanding, the measurements provided by this detector will be correlated in time.

2 A Modern Approach for Information Fusion

A new approach to solve the MTT problem is to jointly detect and track the multiple objects by modeling the scene state as a collection of moving objects. More specifically, we create dynamic and observation models that describe how the targets jointly behave and are observed, respectively. Please refer to [2] for a comprehensive introduction on statistical multisource-multitarget modeling.

Problem statement Using the mathematical formalism, the scene S_n at the discrete time instant n, $n \geq 0$, is modeled as a triple $\langle \mathbf{x}_n^p, \mathbf{M}_n, \mathbf{X}_n \rangle$ in which $\mathbf{x}_n^p$ denotes the robot pose, $\mathbf{M}_n$ the background

[1] A track is an internal entity maintained by the tracking system to uniquely represent a target in the observed scenario. The track entails the target's estimated kinematic state as well as other estimated non-kinematic features.

occupancy grid map and $\mathbf{X}_n$ the multi-target state.

We assume that estimates $\hat{\mathbf{x}}^p_{n|n}$ and $\hat{\mathbf{M}}^p_{n|n}$ are available at each time instant n and are produced separately via a simultaneous self-localization and mapping (SLAM) procedure. In the sequel, we describe an integrated statistical model for jointly detect and estimate the multi-target state $\mathbf{X}_n$ from the sensor measurements $\mathbf{Z}_{0:n} \triangleq \{\mathbf{Z}_0, \ldots, \mathbf{Z}_n\}$ produced by the Velodyne sensor up to time instant n. More precisely, in this work, we seek to calculate the minimum mean squared error (MMSE) estimate $\hat{\mathbf{X}}_{n|n}$ of the multi-target hidden state $\mathbf{X}_n$ conditioned on the available measurements $\mathbf{Z}_{0:n}$.

Multi-target dynamic model Let $\boldsymbol{\mathcal{X}} \subseteq \mathbb{R}^{N_x}$ define a state space of dimensionality N_x. A random finite set (RFS) $\mathbf{X}$ is a random variable that assume values as unordered finite sets of the form $\{\mathbf{x}_1, \ldots, \mathbf{x}_k\}$, with $k \in \mathbb{N}$ and $\mathbf{x}_1, \ldots, \mathbf{x}_k \in \boldsymbol{\mathcal{X}}$. The cardinality of a RFS $\mathbf{X}$ is modeled as a discrete random variable with associated probability mass function (p.m.f.) $\rho(k) = P(\{|\mathbf{X}| = k\})$. Moreover, a RFS is completely specified by its cardinality distribution $\rho(k)$ and a family of symmetric[2] joint distributions $p_k(\mathbf{x}_1, \ldots, \mathbf{x}_k)$ that characterize the distribution of its elements over the state space $\boldsymbol{\mathcal{X}}$ given the cardinality k.

Following this approach, the multi-target state could be modeled as a RFS $\boldsymbol{X}_n \triangleq \{\mathbf{x}^1_n, \ldots, \mathbf{x}^k_n\}$ of $k \triangleq |\boldsymbol{X}_n|$ vehicles states. Furthermore, the multi-target dynamic model, which describes the global state evolution from time instant $n-1$ to n, has a corresponding finite set statistics (FISST), which is a generalization of the p.d.f. for RFS, given by

$$\boldsymbol{X}_n | \boldsymbol{X}_{n-1} \sim f_{n|n-1}(\boldsymbol{X}_n | \boldsymbol{X}_{n-1}) \tag{1}$$

that models both how targets jointly behave over time and also how targets appear and disappear from the scene.

TODO: To formally describe the multi-target dynamic model.

In this work, each vehicle is, in turn, individually modeled as a 3D box with state $\mathbf{x}_n = \begin{bmatrix} x_n & \dot{x}_n & y_n & \dot{y}_n & z_n & \dot{z}_n & w & l & h \end{bmatrix}^T$ consisting of the positions and velocities of the vehicle's centroid respectively in dimensions x, y and z as well as of its size attributes: width w, length l and height h. Since vehicles are rigid bodies, we assume the size attributes are static, i.e. they are not supposed to change over time.

For the sake of simplicity, we also assume that vehicles move independently in a sparsely populated scenario/road according to a version of the white noise acceleration model [1] extended to incorporate the size attributes. In general, this premise does not hold and vehicles' maneuvers could be strongly correlated, e.g. vehicles moving in a crowded urban area. The proposed white noise acceleration model is given as

$$\mathbf{x}_n = \mathbf{F}_n \mathbf{x}_{n-1} + \mathbf{w}_n \tag{2}$$

where $\mathbf{w}_n$ represents a zero-mean independent, identically distributed (i.i.d.) multi-variate Gaussian noise process with known covariance matrix $\mathbf{Q}_n$, i.e. $\mathbf{w}_n \sim \mathcal{N}(\cdot | \mathbf{0}, \mathbf{Q}_n)$.

The state transition matrix parameterized by the interval $\Delta_n = t_n - t_{n-1}$ elapsed between the current and the last sensor scans could be written as

$$\mathbf{F}_n = \begin{bmatrix} \tilde{\mathbf{F}}_n & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{F}}_n & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \tilde{\mathbf{F}}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{3\times 3} \end{bmatrix} \tag{3}$$

[2] A joint distribution function $p_k(\mathbf{x}_1, \ldots, \mathbf{x}_k)$ is symmetric if its value remains unchanged for any of the possible $k!$ permutations of its variables $\mathbf{x}_1, \ldots, \mathbf{x}_k$.

with

$$\tilde{\mathbf{F}}_n = \begin{bmatrix} 1 & \Delta_n \\ 0 & 1 \end{bmatrix},$$

where $\mathbf{0}$ and $\mathbf{I}_{3\times 3}$ denote a zero matrix with appropriate dimensions and a 3-by-3 identity matrix, respectively.

Additionally, the covariance matrix is given by

$$\mathbf{Q} = \begin{bmatrix} \sigma_x^2\tilde{\mathbf{Q}}_n & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sigma_y^2\tilde{\mathbf{Q}}_n & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sigma_z^2\tilde{\mathbf{Q}}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \sigma_{size}^2\mathbf{I}_{3\times 3} \end{bmatrix} \tag{4}$$

with

$$\tilde{\mathbf{Q}}_n = \begin{bmatrix} \Delta_n^3/3 & \Delta_n^2/2 \\ \Delta_n^2/2 & \Delta_n \end{bmatrix},$$

where σ_x, σ_y, σ_z are respectively the acceleration standard deviations at the x, y and z directions and σ_{size} is the size standard deviation for all vehicle dimensions w, l and h.

Note that $\mathbf{F}_n$ and $\mathbf{Q}_n$ are block diagonal matrices, thus, this preliminary dynamic model assumes that the vehicles' dynamics at each direction x, y and z are uncoupled. This is a reasonable approximation if, based on the current tracks' states, the tracking system is required to perform short term predictions only, i.e. if the tracker is processing measurements from a sensor which is relatively fast compared to the vehicles dynamics. However, the proposed model is not well suited for long term predictions, since it may fail to capture eventual maneuvers of the vehicles being tracked. In this case, the long term vehicle dynamics may be best modeled by possible non-linear models.

Generalized likelihood function At each discrete time instant n, $n \geq 0$, and for each interrogated direction represented by the index l, the Velodyne laser range finder produces a set of measurements $\mathbf{Z}_n = \{\mathbf{z}_n^l\}$, $l \in \{1, \ldots, L\}$, where L is the number of sensed directions at each scan. An individual sensor measurement $\mathbf{z}_n^l = \begin{bmatrix} \theta_n^l & \phi_n^l & \rho_n^l \end{bmatrix}^T$ consists of the measured azimuth θ_n^l, elevation ϕ_n^l and range ρ_n^l to a detected reflector at the discrete direction l – relative to the robot pose[3] $\mathbf{x}_n^p$ – in the observed scenario. The detected reflector in turn could belong to either a surface of a vehicle in $\mathbf{X}_n$ or to the background represented by $\mathbf{M}_n$.

$$\begin{aligned} \mathbf{z}_n^l &= \mathbf{g}^l(\mathbf{S}_n) + \boldsymbol{\nu}_n^l \\ &= \begin{bmatrix} \theta_n^l(\mathbf{x}_n^p) \\ \phi_n^l(\mathbf{x}_n^p) \\ \rho_n^l(\mathbf{x}_n^p, \mathbf{M}_n, \mathbf{X}_n) \end{bmatrix} + \boldsymbol{\nu}_n^l \end{aligned} \tag{5}$$

where $\boldsymbol{\nu}_n^l \sim \mathcal{N}(\cdot|\mathbf{0}, \mathbf{R}_n^l)$ and $\mathbf{g}^l(\cdot)$ denotes a family of non-linear vector functions of the scenario state $\mathbf{S}_n$ for each sensed direction l. Note that the measured angles θ_n^l and ϕ_n^l depends strictly on the robot pose $\mathbf{x}_n^p$. On the other hand, the range ρ_n^l is also a non-linear function of the multi-target state

[3] Actually, the origin of Velodyne sensor referential does not necessarily coincide with the vehicles estimated centroid. The calibrated sensor frame offset must also be taken into account.

$\mathbf{X}_n$ and the background $\mathbf{M}_n$. Finally, the covariance matrix $\mathbf{R}^l_n$ could be modeled as a diagonal matrix

$$\mathbf{R}^l_n = \begin{bmatrix} \sigma^2_{\theta^l_n} & 0 & 0 \\ 0 & \sigma^2_{\phi^l_n} & 0 \\ 0 & 0 & \sigma^2_{\rho^l_n} \end{bmatrix} \tag{6}$$

parameterized by the standard deviations $\sigma_{\theta^l_n} = \sigma_{\theta^l_n}(\mathbf{x}^p_n)$, $\sigma_{\phi^l_n} = \sigma_{\phi^l_n}(\mathbf{x}^p_n)$ and $\sigma_{\rho^l_n} = \sigma_{\rho^l_n}(\mathbf{x}^p_n, \mathbf{M}_n, \mathbf{X}_n)$, which are also functions of the scenario state $\mathbf{S}_n$.

In the multi-target framework, we should be able to write the generalized likelihood function as a FISST such that

$$\boldsymbol{Z}_n|\boldsymbol{S}_n \sim g_n(\boldsymbol{Z}_n|\boldsymbol{S}_n) \tag{7}$$

which fully describes how likely measurements represent real targets or the background given the scene state $\boldsymbol{S}_n$.

TODO: To detail the generalized likelihood function.

3 Hierarchical Modeling

In this section, we propose a new approach for tracking extended targets based on low-level measurements.

TODO: To formally describe/model the data at each level of the preliminary architecture.

4 Preliminary Design

Fig. 1 presents the preliminary design architecture in terms of its main components and interfaces (IF).

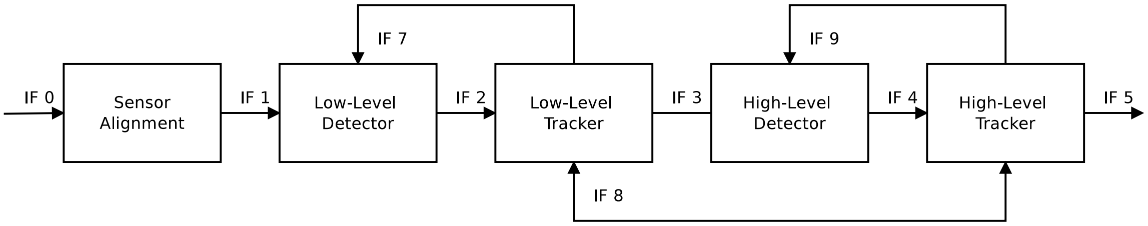

Figure 1: Preliminary design.

In the sequel, we describe the purpose of each component and what is being exchanged via each interface.

- The Sensor Alignment component applies the proper transformations to the Velodyne's raw measurements – received via interface IF 0 – to convert them from spherical to Cartesian coordinates including their corresponding covariance matrices. An unbiased transformation to accomplish this is presented in [1]. The 3D Cartesian points and their covariance matrices are forwarded to the Low-Level Detector via interface IF 1;

- The Low-Level Detector processes the aligned measurements to identify rectangular surfaces which may represent part of an observed vehicle. It delivers a set of detected rectangular surfaces[4] at the interface IF 2;

- The Low-Level Tracker maintains a set of low-level tracks that represent rectangular surfaces being continuously tracked. It updates the low-level tracks based on the detects delivered at IF 2. The low-level tracks are forwarded to the High-Level Detector via the interface IF 3. Note that, despite not necessarily representing a moving object, a low level track is likely to represent a surface of a real obstacle in the world and could already be employed by the obstacle avoidance module. Moreover, the Low-Level Tracker may feedback the low-level tracks to the Low-Level Detector via interface IF 7 to help it pruning unlikely detects or direct its focus on specific areas on which it is more likely to detect rectangular surfaces;

- The High-Level Detector identifies 3D boxes in the observed scenario based on the multiple low-level tracks being reported by the Low-Level Tracker via interface IF 3. More specifically, it identifies plausible combinations of the rectangular surfaces being tracked which are likely to represent a box shaped object in the observed scenario. It delivers the high-level detects at interface IF 4;

- Finally, the High-Level Tracker maintains a set of high-level tracks that represent 3D boxes being tracked and, ultimately, the vehicles in the real world[5]. It employs the high-level detects received via interface IF 4 to update the high-level tracks and delivers the final result at interface IF 5. Similarly to the Low-Level Tracker, the High-Level Tracker could feedback the boxes being tracked to the High-Level Detector via interface IF 9. Additionally, it could also feedback the high-level tracks to the Low-Level Tracker via interface IF 8. In special, the high-level tracks could be splitted into low-level tracks which are then inserted into the Low-Level Tracker. Next, the High-Level Tracker could directly update its tracks based on the corresponding low-level tracks, bypassing thus the High-Level Detector. Conversely, the Low-Level Tracker could employ long-term predictions produced by the High-Level Tracker to improve the measurement-to-track data association procedure performed at the low-level, i.e. the predicted high-level tracks are splitted into low-level tracks and then associated with low-level measurements. Note that the splitted high-level tracks will compete with existing low-level tracks for the low-level measurements during the data association process. Ultimately, if a high-level track consistently represents a object in the scenario, its corresponding low-level tracks will be more likely to associate with the low-level measurements and the competing tracks will be deleted due to starvation.

References

[1] Y. Bar-Shalom and X. Li, *Multitarget-Multisensor Tracking: Principles and Techniques.* YBS, Storrs, CT, 1995.

[2] R. P. Mahler, *Statistical Multisource-Multitarget Information Fusion.* Artech House Inc., Norwood, 2007.

[4] It is also necessary to model the uncertainty and randomness of this kind of measurement.

[5] We can use a simple Moving Target Indicator (MTI) procedure to discard stationary objects which are, in principle, already represented in the background occupancy grid map.