Proc. of IEEE Intelligent Vehicles Symposium (IV), 2021, Nagoya, Japan. DOI: 10.1109/IV48863.2021.9575636

Optimization based 3D Multi-Object Tracking using Camera and Radar Data

Johannes Pöschmann, Tim Pfeifer and Peter Protzel

***Abstract*— Robust and reliable online 3D multi-object tracking is an essential component of autonomous driving. Recent research follows the tracking-by-detection paradigm and focuses mainly on lidar sensors, due to their superior range, resolution and depth accuracy compared to other automotive sensors. This simplifies the challenging data association in crowded urban road scenes, resulting in a predominant status of laser based methods. In contrast, we propose an online 3D multi-object tracker based solely on mono camera images and radar data to promote non-lidar based tracking research. By representing all detections of one frame as a Gaussian mixture model (GMM), we are able to avoid a fixed data association, which may include wrong assumptions. Instead, we assign the GMM to each tracked object and solve the data association implicitly and jointly by estimating the full 3D object tracks in our factor graph based optimization back end. By including all available information from the object detector, our algorithm achieves accurate, robust and reliable tracking results. We conduct real world experiments on the nuScenes tracking data set improving the state-of-the-art for non-lidar based methods from $17.7\,\%$ to $34.1\,\%$ AMOTA.**

I. Introduction

Robust and reliable 3D multi-object tracking is essential for autonomous driving. Most challenging are urban road scenes with multiple dynamic objects in close proximity, since 3D object detectors struggle to detect each object individually (e.g. in a group of pedestrians) and objects occlude each other frequently. The current state-of-the-art in online 3D multi-object tracking approaches this problem by following the tracking-by-detection paradigm [1, 2]. These methods apply a 3D object detector to estimate the location, rotation and size of each object in the scene from the given sensor input. The key challenge is a robust and reliable data association between detected objects and existing tracks, which is commonly solved via the Hungarian algorithm [3, 4]. The performance of these methods is therefore closely coupled to the quality of the given object detections and the accuracy and robustness of the data association. As a result, recent research focuses mainly on lidars, since their far superior object detections compared to other automotive sensors [5, 6, 7], which also simplifies the data association step.

In contrast, cameras and radars are the most common automotive sensors in modern cars, because they are affordable, easily available and small compared to lasers. Therefore, we propose a 3D multi-object tracker which is solely based on camera and radar sensors. Following the common tracking-by-detection paradigm, an of-the-shelf 3D object detector [7] is applied to estimate the position, orientation, size and velocity of all objects in the scene. Although the algorithm fuses mono camera images and radar data for a better depth estimation, the resulting object detections are less accurate and reliable compared to laser based methods [5, 7]. This is critical in situations with multiple dynamic objects in close proximity (e.g. a group of pedestrians), since the data association will most likely contain wrong matchings. We approach this problem by designing a robust optimization back end, which is able to solve the data association implicitly and jointly with the estimation of the full 3D object tracks during inference. All detections of one frame are collectively represented as a single Gaussian mixture model (GMM), including the position, velocity, dimensions and rotation of the 3D bounding boxes from the object detector. The full GMM

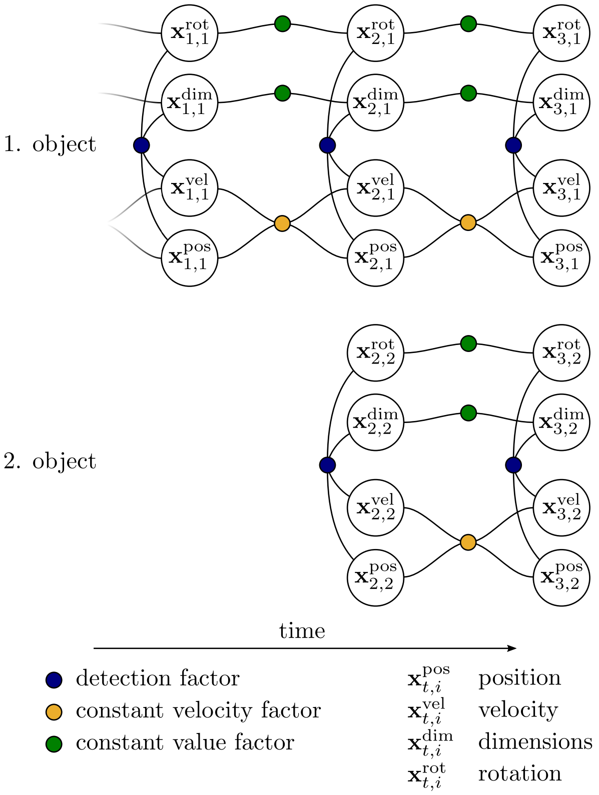
1. object
2. object
$\mathbf{x}_{1,1}^{\mathrm{rot}}$
$\mathbf{x}_{1,1}^{\mathrm{dim}}$
$\mathbf{x}_{1,1}^{\mathrm{vel}}$
$\mathbf{x}_{1,1}^{\mathrm{pos}}$
$\mathbf{x}_{2,1}^{\mathrm{rot}}$
$\mathbf{x}_{2,1}^{\mathrm{dim}}$
$\mathbf{x}_{2,1}^{\mathrm{vel}}$
$\mathbf{x}_{2,1}^{\mathrm{pos}}$
$\mathbf{x}_{3,1}^{\mathrm{rot}}$
$\mathbf{x}_{3,1}^{\mathrm{dim}}$
$\mathbf{x}_{3,1}^{\mathrm{vel}}$
$\mathbf{x}_{3,1}^{\mathrm{pos}}$
$\mathbf{x}_{2,2}^{\mathrm{rot}}$
$\mathbf{x}_{2,2}^{\mathrm{dim}}$
$\mathbf{x}_{2,2}^{\mathrm{vel}}$
$\mathbf{x}_{2,2}^{\mathrm{pos}}$
$\mathbf{x}_{3,2}^{\mathrm{rot}}$
$\mathbf{x}_{3,2}^{\mathrm{dim}}$
$\mathbf{x}_{3,2}^{\mathrm{vel}}$
$\mathbf{x}_{3,2}^{\mathrm{pos}}$
time
● detection factor
● constant velocity factor
● constant value factor
$\mathbf{x}_{t,i}^{\mathrm{pos}}$ position
$\mathbf{x}_{t,i}^{\mathrm{vel}}$ velocity
$\mathbf{x}_{t,i}^{\mathrm{dim}}$ dimensions
$\mathbf{x}_{t,i}^{\mathrm{rot}}$ rotation

Fig. 1. Factor graph representation of the solved estimation problem. Each tracked object is represented as an independent set of states, including its position, velocity, spatial dimensions and rotation. Our multimodal detection factor (blue) assigns all detections simultaneously to all available objects. The actual assignment is done implicitly during the optimization and can change over time.

The authors are with the Faculty of Electrical Engineering and Information Technology, Technische Universität Chemnitz, Germany.
Email: johannes.poeschmann@tu-chemnitz.de, tim.pfeifer@tu-chemnitz.de, peter.protzel@tu-chemnitz.de

is integrated into our factor graph via the Max-Mixture [8] approximation. In combination with a simple motion model, we can estimate the full 3D object tracks without solving the data association explicitly. Furthermore, this association is not fixed and can be changed in future optimization steps with more available information.

The proposed algorithm is an ongoing development of our proof-of-concept for factor graph based multi-object tracking [9] and contributes:

Camera and radar based 3D multi-object tracking: We utilize common automotive sensors for multi-object tracking to work against the recent focus on lidar sensors and to promote camera and radar based methods.

Estimation of the full 3D object tracks: Our algorithm includes all available information from the 3D object detector into the implicit data association and the estimation of the object tracks. Therefore, it can optimize the 3D position, velocity, dimensions and rotation of the tracked objects.

State-of-the-art tracking results: We conduct real world experiments on the nuScenes tracking data set improving the state-of-the-art for non-lidar based methods from $17.7\,\%$ to $34.1\,\%$ AMOTA and reducing the track fragmentation by $36.6\,\%$. We provide a comprehensive ablation study of the utilized data, which shows that including more information into the implicit data association and the state estimation results in more accurate, robust and reliable tracking results. Furthermore, videos and the source code of our algorithm are available online under an open source license [1].

II. Related Work

A. Automotive Object Detection

All state-of-the-art 3D object detectors are neural network based approaches [5, 7, 10]. Recent research focuses mainly on lidar point clouds as input, since they directly provide 3D data with superior range, resolution and depth accuracy compared to other automotive sensors like cameras or radars [5, 10]. Another common approach is the combination of camera and laser based features inside a deep neural network [11, 12]. Obtaining accurate and reliable 3D bounding boxes from camera or radar is a lot more challenging, since the sensors either lack a robust depth estimation or suffient resolution. The authors of [13] use stereo camera images to reconstruct the depth of the scene and to propose 3D object detections. In [14] and [15], an end-to-end learned deep neural network is applied to obtain 3D object detections directly from 2D camera images by estimating the depth solely from mono camera images. The authors of [7] propose a fusion algorithm that associates proposed 3D bounding boxes from mono camera images with radar detections and refines their features in a neural network for a better depth and velocity estimation. We use this algorithm as an out-of-the-box object detector for our 3D multi-object tracking algorithm.

B. Online Multi-Object Tracking

Most state-of-the-art online multi-object trackers follow the tracking-by-detection paradigm [1, 2, 3]. Recent research in the domain of neural network based 3D multi-object tracking focuses mainly on end-to-end learned models like [2, 16], since object detection and tracking are combined in one pipeline and learned simultaneously. The authors of [16] propose a robust neural network based fusion module to merge features from lidar and camera input for better tracking results. In [2], a simultaneous detection and tracking algorithm is proposed, which detects objects as points in temporal image pairs and predicts the association from frame to frame. Another approach is the combination of neural networks and filter based solutions. The authors of [17] use a deep neural network to obtain object detections from 2D images and combine it with a Poisson multi-Bernoulli mixture filter to generate 3D tracks. Other common filter based approaches are particle [18] or Kalman [3, 19] filter. These methods usually implement the data association step as a bipartite graph matching and solve it once every time step via the Hungarian algorithm [3, 4] or a custom greedy approach [19] and assume it to be fixed afterwards. This is critical, if the data association contains wrong matchings. We avoid this problem by formulating an implicit data association, which is solved jointly with the state estimation during inference. This allows our algorithm to correct the data association of past time steps with new information relaxing the problem of wrong matchings.

C. Factor Graphs in Tracking Applications

A common use case of factor graphs in tracking applications is the data association step. The authors of [20] present a factor graph based method to solve the data association for a multi-object tracking problem in a 2D simulation and combine it with an extended Kalman filter to track objects over time. In [21], a factor graph is utilized to solve the data association of a 2D cell tracking problem, but the resulting cell tracks are not optimized. The authors of [22] solve the data association for closely moving objects based on a factor graph and compare it against a JPDA (joint probabilistic data association). However, their experiments are limited to two objects in a 2D simulation.

Factor graph based data fusion is also common in tracking applications. The authors of [23] and [24] use a factor graph to fuse the data of multiple sensors respectively agents and combine it with a particle based belief propagation to track objects over time. In both cases, the experiments are limited to 2D examples in a simulated environment. In [25] the data of multiple sensors and an Extended Kalman Filter are integrated into a single factor graph framework. However, they evaluate the approach only in 2D simulation for a single tracked object. The authors of [26] use a factor graph formulation for joint inference of multi-object tracking and navigation in a multi agent scenario, which is based on a particle filter implementation. Again, the authors only conduct 2D experiments in a simulated environment.

[1] https://github.com/TUC-ProAut/FG-3DMOT

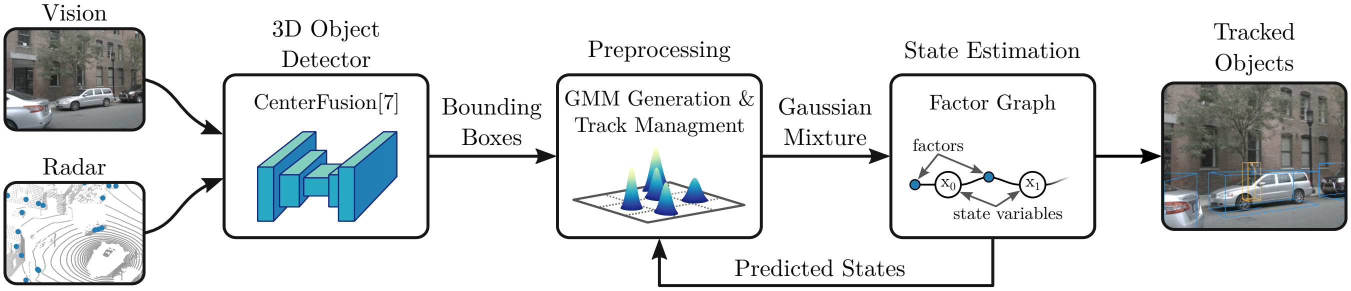

Fig. 2. Data flow of the proposed multi-object tracking algorithm. We represent the objects from the CenterFusion [7] object detector as a multivariate Gaussian mixture model. Beside the position and spatial dimensions of the bounding boxes, we also incorporate the velocity and orientation into the model. During this preprocessing step, the predicted states from the factor graph allow us to decide about creating new tracks or respectively terminating old ones. The association between states and measurements is not fixed and can be changed at any time by optimizing the factor graph.

Our previous work [9] proposed factor graph based 3D multi-object tracking in lidar point clouds. In contrast to this proof-of-concept is our current work centered around the more challenging camera and radar based multi-object tracking. We are able to avoid a fixed data association by representing all detections of one frame as a Gaussian mixture model (GMM) and assigning it to each tracked object. Therefore, the data association is solved implicitly and jointly with the state estimation during optimization of the factor graph. We account for inaccuracies of the object detector by including all available information into the GMM. This also enables our tracking algorithm to optimize the full 3D object tracks. We provide a comprehensive ablation study of the utilized information in the implicit data association and state estimation in Sec. V-C. It proves, that the accuracy, robustness and reliability of the achieved object tracks increase with more available information.

III. Factor Graphs for State Estimation

The most important difference of our approach to the state-of-the-art is the formulation of the optimization problem as factor graph. Combined with a powerful Gaussian mixture model, this allows us to track objects without a fixed assignment between measurements and states.

A. Graphs and Least Squares

We formulate the tracking of objects in 3D space as a probabilistic state estimation problem, which includes the position, velocity, size and rotation of each object, combined in set of states $\mathbf{X}$. It can be formulated as the search for optimal set of states $\mathbf{X}^{*}$, given a set of Measurements $\mathbf{Z}$:

$$\mathbf{X}^{*} = \underset{\mathbf{X}}{\operatorname{argmax}}\, \mathbf{P}(\mathbf{X}|\mathbf{Z}) \tag{1}$$

Under the assumption of Gaussian distributed noise, we can convert the maximization of probabilities to a minimization of negative log-likelihoods in

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmin}} \sum_{i} \frac{1}{2} \left\| \boldsymbol{\mathcal{I}}^{\frac{1}{2}} \left(\mathbf{e}_i - \boldsymbol{\mu} \right) \right\|^{2}. \tag{2}$$

This maximum-likelihood estimator $\hat{\mathbf{X}}$ can be obtained by applying non-linear least squares optimization of the residual function $\mathbf{e}_i = f(\mathbf{x}_i, \mathbf{z}_i)$. We use mean $\boldsymbol{\mu}$ and square root information matrix $\boldsymbol{\mathcal{I}}^{\frac{1}{2}}$ of the Gaussian distribution to describe the measurements error characteristics. Furthermore, we can describe the estimation problem as factor graph, as shown in Fig. 1, to visualize its structure.

Since the least squares approach can be efficiently applied to large sets of states and measurements, we keep the whole trajectory inside the factor graph. This enables our tracking algorithm to improve the estimation of past states with future information resulting in refined estimations of the 3D object tracks at the current time step.

B. Implicit Assignment

The data association is the most crucial step in tracking-by-detection algorithms, since a consistent tracking requires correct assignments of incoming measurements to the existing states. Performing a fixed assignment based on predicted states is an error-prone workaround that is often used in state-of-the-art approaches [1, 3]. Since there is only a coarse knowledge about the true object position, it is likely that wrong assignments occur. With a fixed scheme, this misassignment can never be corrected.

We can avoid problems by using the implicit assignment scheme, which we have proposed in [9]. Hence, we describe the whole set of measurements with an equally weighted Gaussian mixture model (GMM)

$$\begin{gathered}\mathbf{P}\left(\mathbf{z}_i|\mathbf{x}_i\right) \propto \sum_{j=1}^{n} c_j \cdot \exp\left(-\frac{1}{2}\left\| \boldsymbol{\mathcal{I}}_j^{\frac{1}{2}} \left(\mathbf{e}_i - \boldsymbol{\mu}_j\right)\right\|^{2}\right), \\ \text{with } c_j = w_j \cdot \det\left(\boldsymbol{\mathcal{I}}_j^{\frac{1}{2}}\right).\end{gathered} \tag{3}$$

By equally assigning each measurement $\mathbf{z}_j$ with mean $\boldsymbol{\mu}_j$ and uncertainty $\boldsymbol{\mathcal{I}}_j^{\frac{1}{2}}$ to each state $\mathbf{x}_i$, we do not include any assumption regarding the data association. Instead, the assignment can be solved implicitly and jointly with the state estimation during least squares optimization by combining all available information. This data assignment is not fixed and can be changed during future inference steps, similar to the re-estimation of past states.

Using Gaussian mixtures in (2) of course breaks the Gaussian assumption, that leads to the efficient least squares formulation. The authors of [8] propose an effective solution by approximating the sum inside (3) with a maximum-operator, which allows us to represent the GMM inside a least squares problem. A detailed explanation of the Max-Mixture approach can be found in the original publication [8] or our previous work [27]. The actual error functions are described in the next section.

IV. Factor Graph based Tracking

An overview of our online 3D multi-object tracking algorithm is given in Fig. 2 and the whole algorithm is shown in Alg. 1. We obtain 3D object detections solely from mono camera images and radar data, which is explained in Sec. IV-A. Subsequently, all detections of one frame are jointly represented as a GMM, which is incorporated in our factor graph back end besides other constraints of the state estimation. The structure of the factor graph is explained in Sec. IV-B and visualized in Fig. 1. We explain the estimation of the full 3D object tracks and the necessary track management in Sec. IV-C.

A. Object Detection

We use mono camera images and radar data as input for an out-of-the-box 3D object detector [7] at each time step t to obtain 3D bounding boxes $\mathbf{z}_{t,j}$ defined by their center point $\mathbf{z}^{\text{pos}}_{t,j}$, velocity $\mathbf{z}^{\text{vel}}_{t,j}$, dimension $\mathbf{z}^{\text{dim}}_{t,j}$, rotation $\mathbf{z}^{\text{rot}}_{t,j}$, class $z^{\text{class}}_{t,j}$ and confidence $z^{\text{conf}}_{t,j}$:

$$\begin{aligned} \mathbf{z}_{t,j} &= \left[\mathbf{z}^{\text{pos}}_{t,j}, \mathbf{z}^{\text{vel}}_{t,j}, \mathbf{z}^{\text{dim}}_{t,j}, \mathbf{z}^{\text{rot}}_{t,j}, z^{\text{conf}}_{t,j}, z^{\text{class}}_{t,j}\right] \\ \mathbf{z}^{\text{geo}}_{t,j} &= \left[\mathbf{z}^{\text{pos}}_{t,j}, \mathbf{z}^{\text{vel}}_{t,j}, \mathbf{z}^{\text{dim}}_{t,j}, \mathbf{z}^{\text{rot}}_{t,j}\right] \\ \mathbf{z}^{\text{pos}}_{t,j} &= \left[z^{x}_{t,j}, z^{y}_{t,j}, z^{z}_{t,j}\right], \ \mathbf{z}^{\text{vel}}_{t,j} = \left[z^{v_x}_{t,j}, z^{v_y}_{t,j}, z^{v_z}_{t,j}\right] \\ \mathbf{z}^{\text{dim}}_{t,j} &= \left[z^{w}_{t,j}, z^{l}_{t,j}, z^{h}_{t,j}\right], \ \mathbf{z}^{\text{rot}}_{t,j} = \left[z^{\cos\theta}_{t,j}, z^{\sin\theta}_{t,j}\right] \end{aligned} \tag{4}$$

The rotation $\mathbf{z}^{\text{rot}}_{t,j}$ encodes the yaw angle of the 3D bounding box and is defined on the unit circle for a continuous and unambiguous rotation between 0° and 360°. We introduce confidence threshold c_{det} to filter out all detections with $z^{\text{conf}}_{t,j} < c_{\text{det}}$, since most of them are false positives. Besides that, the object detector outputs a lot of overlapping bounding boxes for the same physical object. Therefore, we delete all boxes which overlap more than b_{overlap} with another bounding box of the same class and a higher confidence.

B. States and Factors

We utilize a separated factor graph for each class including the following estimated states for each tracked object obj_i at each time step t :

$$\begin{aligned} \mathbf{x}^{\text{all}}_{t,i} &= \left[\mathbf{x}^{\text{pos}}_{t,i}, \mathbf{x}^{\text{vel}}_{t,i}, \mathbf{x}^{\text{dim}}_{t,i}, \mathbf{x}^{\text{rot}}_{t,i}\right] \\ \mathbf{x}^{\text{pos}}_{t,i} &= \left[p^{x}_{t,i}, p^{y}_{t,i}, p^{z}_{t,i}\right], \ \mathbf{x}^{\text{vel}}_{t,i} = \left[v^{x}_{t,i}, v^{y}_{t,i}, v^{z}_{t,i}\right] \\ \mathbf{x}^{\text{dim}}_{t,i} &= \left[d^{w}_{t,i}, d^{l}_{t,i}, d^{h}_{t,i}\right], \ \mathbf{x}^{\text{rot}}_{t,i} = \left[r^{\cos\theta}_{t,i}, r^{\sin\theta}_{t,i}\right] \end{aligned} \tag{5}$$

State $\mathbf{x}^{\text{pos}}_{t,i}$ encodes the 3D position, $\mathbf{x}^{\text{vel}}_{t,i}$ the 3D velocity, $\mathbf{x}^{\text{dim}}_{t,i}$ the 3D bounding box size and $\mathbf{x}^{\text{rot}}_{t,i}$ its yaw angle on the unit circle.

By applying the Max-Mixture [8] approximation to the sum in (5) we can formulate the following error function for the detection factor $\mathbf{e}^{\text{det}}_{t,i}$ which incorporates the GMM into the factor graph:

$$\begin{aligned} \left\|\mathbf{e}^{\text{det}}_{t,i}\right\|^2 &= \min_j \left\| \begin{matrix} \sqrt{-2 \cdot \ln \frac{c_{t,j}}{\gamma_m}} \\ \boldsymbol{\mathcal{I}}^{\frac{1}{2}}_j \left(\mathbf{e}_{t,i} - \boldsymbol{\mu}_{t,j}\right) \end{matrix} \right\|^2 \\ \text{with } \gamma_m &= \max_j c_{t,j}, \ \boldsymbol{\mathcal{I}}^{\frac{1}{2}}_j = \left(\boldsymbol{\Sigma}^{\text{det}}\right)^{-\frac{1}{2}} \\ \mathbf{e}_{t,i} &= \mathbf{x}^{\text{all}}_{t,i}, \ \boldsymbol{\mu}_{t,j} = \mathbf{z}^{\text{geo}}_{t,j} \end{aligned} \tag{6}$$

We enable an implicit data association by adding the detection factor $\mathbf{e}^{\text{det}}_{t,i}$ to all estimated states. A generic null-hypothesis with mean $\boldsymbol{\mu}_{t,0} = \text{mean}\left(\boldsymbol{\mu}_{t,j} \ \forall \ j\right)$ and a broad uncertainty $\boldsymbol{\Sigma}^{\text{det}}_0$ is added to the GMM for robustness against errors of the object detector. We utilize a simple constant velocity model connecting the $\mathbf{x}^{\text{pos}}_{t,i}$ and $\mathbf{x}^{\text{vel}}_{t,i}$ states of one object over time to describe its movement and to predict existing tracks into the future:

$$\left\|\mathbf{e}^{\text{cv}}_{t,i}\right\|^2_{\boldsymbol{\Sigma}^{\text{cv}}} = \left\| \begin{matrix} \left(\mathbf{x}^{\text{pos}}_{t,i} - \mathbf{x}^{\text{pos}}_{t+1,i}\right) - \mathbf{x}^{\text{vel}}_{t,i} \cdot \Delta t \\ \mathbf{x}^{\text{vel}}_{t,i} - \mathbf{x}^{\text{vel}}_{t+1,i} \end{matrix} \right\|^2_{\boldsymbol{\Sigma}^{\text{cv}}} \tag{7}$$

To enforce a fixed 3D bounding box size for each obj_i, a constant value factor is added between states $\mathbf{x}^{\text{dim}}_{t-1,i}$ and $\mathbf{x}^{\text{dim}}_{t,i}$:

$$\left\|\mathbf{e}^{\text{cd}}_{t,i}\right\|^2_{\boldsymbol{\Sigma}^{\text{cd}}} = \left\|\mathbf{x}^{\text{dim}}_{t-1,i} - \mathbf{x}^{\text{dim}}_{t,i}\right\|^2_{\boldsymbol{\Sigma}^{\text{cd}}} \tag{8}$$

In the same way, a stable rotation $\mathbf{x}^{\text{rot}}_{t,i}$ is ensured by

$$\left\|\mathbf{e}^{\text{cr}}_{t,i}\right\|^2_{\boldsymbol{\Sigma}^{\text{cr}}} = \left\|\mathbf{x}^{\text{rot}}_{t-1,i} - \mathbf{x}^{\text{rot}}_{t,i}\right\|^2_{\boldsymbol{\Sigma}^{\text{cr}}} . \tag{9}$$

The structure of the resulting factor graph is visualized for a generic example with two tracked objects in Fig. 1.

C. Track Management and State Estimation

As mentioned earlier, an independent factor graph is utilized for each class for simplicity and to separate measurements of different classes in our implicit data association. An algorithmic overview of our approach is given in Alg. 1 and the track management and the estimation of the full 3D object tracks is explained in detail in this section.

At the first frame of a sequence, a new object obj_i is initialized with $\mathbf{x}^{\text{all}}_{t,i} = \mathbf{z}^{\text{geo}}_{t,j}$ for each measurement. The motion of the objects can be predicted into the future with the help of (7) after adding all necessary factors and the optimization of the factor graph. Since we do not use an explicit data association step, we cannot find matchings between detections and predicted tracks in order to initialize, continue and terminate tracks. Therefore, we utilize an auxiliary data association based on the similarity of objects and detections in order to find correspondences between objects and measurements needed for track management. We create a similarity matrix between all $\mathbf{x}^{\text{all}}_{t,i}$ (column) and $\mathbf{z}^{\text{geo}}_{t,j}$ (row), including null-hypothesis z_0, based on $-\log\left(\mathbf{P}\left(\mathbf{z}^{\text{geo}}_{t,j} | \mathbf{x}^{\text{all}}_{t,i}\right)\right)$. By finding the minimum (best similarity) we get the correspondence between an object $obj_{t,i}$ and a measurement $\mathbf{z}_{t,j}$ and delete the respective column and row from the matrix, except for the row of the null-hypothesis z_0. In this way, all objects without a correspondence to a real measurement are matched with the null-hypothesis. This step is repeated until all correspondences

Algorithm 1: Online Tracking Algorithm

```
generate detections $\mathbf{Z}$ using CenterFusion [7]
delete all $\mathbf{z}_{t,j}$ with $z^{\text{conf}}_{t,j} < c_{\text{det}}$
delete overlapping $\mathbf{z}_t$ based on $b_{\text{overlap}}$ and $z^{\text{conf}}_{t,j}$
foreach class $c$ do
    initialize factor graph
    foreach time step $t$ do
        create GMM based on $\mathbf{z}^{\text{geo}}_{t,j}$ with $z^{\text{class}}_{t,j} = c$ and
          null-hypothesis $z_0$
        if $t == 0$ then
            init $obj_{t,i}$ with $\mathbf{x}^{\text{all}}_{t,i} = \mathbf{z}^{\text{geo}}_{t,j}$
        else
            propagate $\mathbf{x}^{\text{all}}_{t-1,i}$ to $t$
            get correspondence between $obj_{t,i}$ and $\mathbf{z}_{t,j}$
              according to $-\log\left(\mathbf{P}\left(\mathbf{z}^{\text{geo}}_{t,j}|\mathbf{x}^{\text{all}}_{t,i}\right)\right)$
            if $obj_{t,i}$ not corresponds to any $\mathbf{z}_{t,j}$ then
                mark $obj_{t,i}$ as lost and delete if
                  $n_{\text{lost}} > n_{\text{keep}}$
            end
            if $\mathbf{z}_{t,j}$ not corresponds to any $obj_{t,i}$ then
                init $obj_{t,i}$ with $\mathbf{x}^{\text{all}}_{t,i} = \mathbf{z}^{\text{geo}}_{t,j}$
            end
        end
        add factors (6), (7), (8) and (9)
        optimize factor graph
        output all $obj_{t,i}$ with $n_{\text{lost}} <= n_{\text{perm}}$
    end
end
```

are solved. For each measurement without a matching object, a new track $obj_{t,i}$ is initialized at $\mathbf{x}^{\text{all}}_{t,i} = \mathbf{z}^{\text{geo}}_{t,j}$. All objects corresponding to the null-hypothesis are marked as lost and terminated after more than n_{keep} consecutive time steps without a matching to a real measurement. Our algorithm outputs object tracks starting from the first detection and for n_{perm} time steps after they are marked as lost. The correspondences between objects $obj_{t,i}$ and measurements $\mathbf{z}_{t,j}$ are only used for track management and not for the implicit data association, which is solved entirely during optimization.

After all tracks are managed we add factors (6), (7), (8) and (9) and optimize the factor graph at each time step t. This solves the data association implicitly and jointly with the estimation of the full 3D object tracks by combining all available information of the entire factor graph. This approach has three major advantages: Our algorithm is robust against inaccuracies of the object detector by solving the data association during inference, it can correct inaccurate detections by optimizing the full 3D bounding boxes for the tracked objects and past states can be re-assign and re-estimated with future information, resulting in refined 3D object tracks for the current time step t. We provide a comprehensive analysis of the importance of position $\mathbf{x}^{\text{pos}}_{t,i}$, velocity $\mathbf{x}^{\text{vel}}_{t,i}$, dimension $\mathbf{x}^{\text{dim}}_{t,i}$ and rotation $\mathbf{x}^{\text{rot}}_{t,i}$ information for our implicit data association and state estimation in section Sec. V-C.

TABLE I
PARAMETERS OF OUR ALGORITHM

Parameter Name	**Symbol**	**Value**
Detection Covariance	$\mathbf{\Sigma}^{\text{det}}$	$\operatorname{diag}\begin{pmatrix}0.1\,\mathrm{m}\\0.1\,\mathrm{m}\\0.1\,\mathrm{m}\\1\,\mathrm{m\,s^{-1}}\\1\,\mathrm{m\,s^{-1}}\\1\,\mathrm{m\,s^{-1}}\\0.5\,\mathrm{m}\\0.5\,\mathrm{m}\\0.5\,\mathrm{m}\\1\\1\end{pmatrix}^2$
Detection Null-Hypothesis Covariance	$\mathbf{\Sigma}^{\text{det}}_0$	$\mathbf{\Sigma}^{\text{det}} \cdot 10^4$
Constant Velocity Covariance	$\mathbf{\Sigma}^{\text{cv}}$	$\operatorname{diag}\begin{pmatrix}0.1\,\mathrm{m}\\0.1\,\mathrm{m}\\0.1\,\mathrm{m}\\0.5\,\mathrm{m\,s^{-1}}\\0.5\,\mathrm{m\,s^{-1}}\\0.5\,\mathrm{m\,s^{-1}}\end{pmatrix}^2$
Constant Box Dimension Covariance	$\mathbf{\Sigma}^{\text{cd}}$	$\operatorname{diag}\begin{pmatrix}1\,\mathrm{m}\\1\,\mathrm{m}\\1\,\mathrm{m}\end{pmatrix}^2$
Constant Rotation Covariance	$\mathbf{\Sigma}^{\text{cr}}$	$\operatorname{diag}\begin{pmatrix}1\\1\end{pmatrix}^2$
Detection Confidence Threshold	c_{det}	10 %
Bounding Box Overlap Threshold	b_{overlap}	15 %
Num. Con. Null-Hypothesis Detections	n_{keep}	2
Object Permanence	n_{perm}	1

V. Experiments

A. Setup

We conduct real world experiments on the challenging nuScenes data set [29]. It features 700 training and 150 validation and test scenes in an urban road environment. The data set includes a laser scanner, 6 cameras with a 360° view around the ego vehicle and 5 radar sensors. Each scene has a length of 20 s and is annotated with 3D bounding boxes at a frame rate of approximately 2 Hz, resulting in 40 key frames per scene. The object classes bicycle, bus, car, motorcycle, pedestrian, trailer and truck are evaluated separately for the tracking task. The two most important metrics are the average multi-object tracking accuracy (AMOTA) and precision (AMOTP) with a 40 point interpolation over different recall thresholds to approximate integral metrics for a better evaluation [3]:

$$\text{AMOTA} = \frac{1}{n-1} \sum_{r \in \{\frac{1}{n-1}, \frac{2}{n-1}, \ldots, 1\}} \text{MOTAR} \tag{10}$$

$$\text{MOTAR} = \max\left(0, 1 - \frac{IDS_r + FP_r + FN_r - P \cdot (1-r)}{rP}\right) \tag{11}$$

$$\text{AMOTP} = \frac{1}{n-1} \sum_{r \in \{\frac{1}{n-1}, \frac{2}{n-1}, \ldots, 1\}} \frac{\sum_{i,t} d_{i,t}}{\sum_t TP_t} \tag{12}$$

TABLE II

RESULTS ON THE NUSCENES TRACKING TEST SET

Method	**Modalities**	**AMOTA** ↑	**AMOTP** ↓	**MOTAR** ↑	**Recall** ↑	**TP** ↑	**FP** ↓	**IDS** ↓	**FRAG** ↓
DEFT [28]	Camera	17.7 %	1.57 m	48.4 %	33.8 %	52 099	22 163	6901	3420
DBNet*	Camera	7.2 %	1.49 m	34.9 %	20.2 %	33 819	19 874	3831	2726
ProTracker*	Camera	7.2 %	1.63 m	25.6 %	23.9 %	32 245	**17 957**	4006	2935
Buffalo_Vision*	Camera	5.9 %	1.49 m	24.4 %	24.3 %	32 325	26 003	4170	3157
CaTracker*	Camera	5.3 %	1.61 m	16.8 %	29.0 %	33 937	24 574	4457	3169
FG-3DMOT (ours)	Camera, Radar	**34.1 %**	**1.25 m**	**60.7 %**	**43.7 %**	**61 893**	18 110	**2032**	**1728**

*Methods without publications.

AMOTA includes false positives FP_r, false negatives FN_r and id switches IDS_r. P denotes the ground truth positives and $P\,(1-r)$ is used for recall normalization. The AMOTP metric summarizes the spatial error of the tracks. $d_{i,t}$ denotes the position error of track i at time t, which is divided by the number of matches TP_t.

As mentioned earlier, we use CenterFusion [7] as an out-of-the-box object detector based on mono camera images and radar data. It provides 3D bounding boxes, velocities in x and y direction and a confidence for each object detection at a framerate of 2 Hz. All other parameters of our approach are summarized in Tab. I. We use libRSF [30] as our factor graph back end and solve the least squares optimization with Ceres [31].

B. Results

We evaluated our online 3D multi-object tracker on the nuScenes tracking benchmark server. The results of our algorithm and the best five non-lidar based approaches from the leaderboard are summarized in Tab. II (accessed February 2021). Please note, that most of the methods from the leaderboard do not provide a scientific publication, since non-lidar based 3D multi-object tracking algorithms are highly under-represented on the nuScenes dataset. Furthermore, our approach is the first entry based on camera and radar data.

Our proposed algorithm improves the state-of-the-art for tracking without lidar by considerable 16.4 % from 17.7 % to 34.1 % AMOTA. Additionally, we are able to half the number of id switches (IDS) and achieve a 36.6 % lower track fragmentation (FRAG) compared to the other methods, proving the effectiveness of our combined implicit data association and state estimation. By including all available information into the factor graph we achieve robust, reliable and accurate tracking results. Our approach also improves the precision of the generated tracks (AMOTP) from 1.49 m to 1.25 m by estimating the full 3D object tacks, which compensates for inaccuracies of the object detector. All in all, the accuracy, robustness and reliability of the tracking results of our proposed algorithm are far superior compared to other camera and radar based methods.

C. Ablation Study

We provide a comprehensive ablation study on the information used in the implicit data association and the state estimation to analyze the impact of position $\mathbf{x}_{t,i}^{\text{pos}}$, velocity $\mathbf{x}_{t,i}^{\text{vel}}$, dimension $\mathbf{x}_{t,i}^{\text{dim}}$ and rotation $\mathbf{x}_{t,i}^{\text{rot}}$ data on the robustness of our tracking algorithm. The structure of the factor graph described in Sec. IV does not need to be altered for the ablation study. Instead, we can just exclude information from the implicit data association and state estimation and define idle states as constant. We always utilize the 3D position information and analyze it with all possible combinations of velocity, 3D bounding box dimension and rotation based on the AMOTA. The results of our ablation study are summarized in Tab. III.

The more information we add in the implicit data association and the state estimation the more robust tracking results our algorithm achieves, which is evident for class trailer. It's AMOTA improves from 6.20 % based only on position information to 13.16 % when all available information $\mathbf{z}_{t,j}^{\text{geo}}$ is utilized. Additionally, the average AMOTA over all classes improves from 32.43 % to 36.36 % by including all available information. In contrast, the accuracy of class car does not improve considerably because the 3D position of the detections is reliable on its own. Adding more information does not always improve the AMOTA for single classes. The accuracy of class motorcycle decreases slightly from 34.85 % to 34.27 % with added velocity, while the added information is beneficial for car or pedestrian. This illustrates, that more information in the data association and state estimation results in a more balanced tracking of all classes.

Our algorithm achieves real time feasible frames per second (FPS) for all combinations of utilized information, which is shown in the last column of Tab. III. Adding velocity information into the implicit data association and the state estimation provides the constant velocity model with a robust velocity estimation instead of initializing it with zero. This results in a much faster optimization, demonstrated by the achieved 15.9 FPS for the combination of position and velocity in comparison to the 5.8 FPS for position only. Apart from the velocity, additional information increases the computational load during optimization, which results in a trade off between the quality of the achieved results and the required computational speed. The ablation study proves that the robustness, reliability and accuracy of the tracking results increases with more added information. Furthermore, additional data can be easily integrated into the algorithm due to flexible structure of factor graphs.

TABLE III

Ablation Study on the nuScenes Tracking Validation Set based on AMOTA of the classes and FPS

Used Information	Average	Bicycle	Bus	Car	Motorcycle	Pedestrian	Trailer	Truck	FPS
Position	32.43 %	21.57 %	43.76 %	59.68 %	31.15 %	38.71 %	6.20 %	25.94 %	5.77
Position, Rotation	34.23 %	23.19 %	46.48 %	61.34 %	33.42 %	39.97 %	6.63 %	28.57 %	3.52
Position, Dimension	34.72 %	23.38 %	47.18 %	61.55 %	33.64 %	40.28 %	8.48 %	28.52 %	2.85
Position, Velocity	35.27 %	24.53 %	47.27 %	62.22 %	33.19 %	42.25 %	7.90 %	29.55 %	**15.86**
Pos, Dim, Rot	35.32 %	23.95 %	46.75 %	61.22 %	**34.85 %**	40.34 %	9.97 %	30.14 %	2.11
Pos, Vel, Rot	35.90 %	23.92 %	**47.35 %**	62.82 %	34.11 %	42.24 %	10.57 %	30.31 %	12.10
Pos, Vel, Dim	35.91 %	24.00 %	46.80 %	**62.86 %**	33.86 %	**42.39 %**	10.65 %	30.78 %	9.66
Pos, Vel, Dim, Rot	**36.36 %**	**24.57 %**	46.10 %	62.76 %	34.27 %	42.17 %	**13.16 %**	**31.53 %**	6.44

VI. Conclusion

Our main contributions are an online 3D multi-object tracking algorithm without an explicit data association and the analysis of its core component – the Gaussian mixture model that represents all available detections.

The robustness of this approach stems from an implicit data association that is done jointly with the 3D track optimization. A robust factor graph back end allows us to solve both tasks simultaneously and under real time conditions. By including the combined information from a camera and radar based object detector, our algorithm is able to compensate for its inaccuracies and to refine the 3D bounding boxes. As a result, our proposed method achieves accurate, reliable and robust tracking results based solely on mono camera and radar data. We conducted real world experiments on the nuScenes tracking data set improving the state-of-the-art for tracking without lidar data from 17.7 % to 34.1 % AMOTA. Furthermore, we could reduce the number of id switches by considerable 47.0 % and the track fragmentation by 36.6 %. These results prove the effectiveness of our implicit data association and full 3D state estimation for low quality object detections. Our ablation study of the utilized information inside the factor graph shows, that including more information results in more accurate and robust tracking results. We could also show that an increasing number of information does not necessarily come with an increasing run time. The advantage of velocity measurements is a strong motivation for further research in radar based tracking. Due to the flexible structure of factor graphs, the proposed algorithm can be easily extended by including additional information in the implicit data association, utilizing more sensor modalities or implementing a more sophisticated motion model.

References

[1] H. kuang Chiu, J. Li, R. Ambrus, and J. Bohg, "Probabilistic 3d multi-modal, multi-object tracking for autonomous driving," *CoRR*, vol. abs/2012.13755, 2020.

[2] X. Zhou, V. Koltun, and P. Krähenbühl, "Tracking objects as points," in *Proc. of European Conf. on Computer Vision (ECCV)*, 2020.

[3] X. Weng, J. Wang, D. Held, and K. Kitani, "3d multi-object tracking: A baseline and new evaluation metrics," *CoRR*, vol. abs/1907.03961, 2020.

[4] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. of Intl. Conf. on Image Processing (ICIP)*, 2017.

[5] T. Yin, X. Zhou, and P. Krähenbühl, "Center-based 3d object detection and tracking," *CoRR*, vol. abs/2006.11275, 2021.

[6] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "Pv-rcnn: Point-voxel feature set abstraction for 3d object detection," in *Proc. of Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[7] R. Nabati and H. Qi, "CenterFusion: Center-based Radar and Camera Fusion for 3D Object Detection," *CoRR*, vol. abs/2011.04841, 2020.

[8] E. Olson and P. Agarwal, "Inference on networks of mixtures for robust robot mapping," in *Proc. of Robotics: Science and Systems (RSS)*, Sydney, Australia, 2012.

[9] J. Pöschmann, T. Pfeifer, and P. Protzel, "Factor graph based 3d multi-object tracking in point clouds," in *Proc. of Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[10] S. Shi, X. Wang, and H. Li, "Pointrcnn: 3d object proposal generation and detection from point cloud," in *Proc. of Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[11] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, "Joint 3d proposal generation and object detection from view aggregation," in *Proc. of Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2018.

[12] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," in *Proc. of Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[13] 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 *Proc. of Advances in Neural Information Processing Systems (NIPS)*, 2015.

[14] J. Ku, A. D. Pon, and S. L. Waslander, "Monocular 3d object detection leveraging accurate proposals and shape reconstruction," in *Proc. of Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[15] B. Xu and Z. Chen, "Multi-level fusion based 3d object detection from monocular images," in *Proc. of Intl. Conf.*

on Computer Vision and Pattern Recognition (CVPR), 2018.

[16] W. Zhang, H. Zhou, S. Sun, Z. Wang, J. Shi, and C. C. Loy, “Robust multi-modality multi-object tracking,” in *Proc. of Intl. Conf. on Computer Vision (ICCV)*, 2019.

[17] S. Scheidegger, J. Benjaminsson, E. Rosenberg, A. Krishnan, and K. Granström, “Mono-camera 3d multi-object tracking using deep learning detections and pmbm filtering,” in *Proc. of Intelligent Vehicles Symposium (IV)*, 2018.

[18] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool, “Robust tracking-by-detection using a detector confidence particle filter,” in *Proc. of Intl. Conf. on Computer Vision (ICCV)*, 2009.

[19] H. kuang Chiu, A. Prioletti, J. Li, and J. Bohg, “Probabilistic 3d multi-object tracking for autonomous driving,” *CoRR*, vol. abs/2001.05673, 2020.

[20] H. Wang, J. Sun, S. Lu, and S. Wei, “Factor graph aided multiple hypothesis tracking,” *Science China Information Sciences*, vol. 56, 2013.

[21] M. Schiegg, P. Hanslovsky, C. Haubold, U. Koethe, L. Hufnagel, and F. A. Hamprecht, “Graphical model for joint segmentation and tracking of multiple dividing cells,” *Bioinformatics*, vol. 31, no. 6, 2014.

[22] Viji Paul Panakkal and R. Velmurugan, “Effective data association scheme for tracking closely moving targets using factor graphs,” in *Proc. of Nat. Conf. on Communications (NCC)*, 2011.

[23] F. Meyer, P. Braca, P. Willett, and F. Hlawatsch, “A scalable algorithm for tracking an unknown number of targets using multiple sensors,” *IEEE Transactions on Signal Processing*, vol. 65, no. 13, 2017.

[24] F. Meyer, O. Hlinka, H. Wymeersch, E. Riegler, and F. Hlawatsch, “Distributed localization and tracking of mobile networks including noncooperative objects,” *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 1, 2016.

[25] M. Cheng, M. R. K. Aziz, and T. Matsumoto, “Integrated factor graph algorithm for doa-based geolocation and tracking,” *IEEE Access*, vol. 8, 2020.

[26] F. Meyer and M. Z. Win, “Joint navigation and multi-target tracking in networks,” in *Proc. of Intl. Conf. on Communications Workshops (ICC Workshops)*, 2018.

[27] T. Pfeifer and P. Protzel, “Expectation-maximization for adaptive mixture models in graph optimization,” in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2019.

[28] M. Chaabane, P. Zhang, J. R. Beveridge, and S. O’Hara, “Deft: Detection embeddings for tracking,” *CoRR*, vol. abs/2102.02267, 2021.

[29] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, “nuscenes: A multimodal dataset for autonomous driving,” *CoRR*, vol. abs/1903.11027, 2020.

[30] T. Pfeifer and Others, “libRSF,” https://github.com/TUC-ProAut/libRSF.

[31] S. Agarwal, K. Mierle, and Others, “Ceres Solver,” http://ceres-solver.org.