2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

MOTCues: 3D Multi-Object Tracking with Birth Prior and Shape Description Informed by Point Cloud Cues

Hanyeol Lee[1], Yeongkwon Choe[2], Taeyoon Kim[1], and Chan Gook Park[1]

***Abstract*— Reliable multi-object tracking (MOT) is essential for autonomous systems but remains challenging due to ambiguous object characteristics such as birth, death, and motion models, as well as detector errors including false detections and missed objects. Random finite set (RFS) theory provides a rigorous mathematical foundation that enables the formulation of fundamental uncertainties in object estimation under the Bayesian framework. We propose MOTCues, a MOT algorithm built on the RFS-based Poisson multi-Bernoulli filter, which integrates informative components derived from point cloud cues into the estimator as a tailored formulation. The object birth intensity function is modeled with a Gaussian mixture distribution for effective initialization of new-born objects, while object shape information is captured by constructing bounding box-centric descriptors to enhance hypothesis management. Evaluations on the KITTI dataset and the nuScenes benchmark demonstrate that integrating point cloud cues improves tracking performance by reducing ID switches, achieving superior results compared to baseline model-based trackers in real-world object tracking scenarios.**

I. Introduction

For the safe operation of autonomous vehicles and mobile robots, semantic understanding of the surrounding environment is essential for decision making, including obstacle avoidance and efficient path planning. Intelligent platforms are typically deployed in crowded areas with high demand, where numerous surrounding agents act as moving objects, making the aforementioned challenges even more difficult. Currently, highly advanced data-driven object detectors are driving the next wave of spatial understanding, yet they inevitably produce false positives and missed detections that impede reliable operation of autonomous systems. Track-by-detection-based object trackers estimate the state of each object from detection results by incorporating the time domain, which enables consistent tracking even under temporary occlusions and allows robust rejection of false alarms [1].

Multi-object tracking (MOT) has been actively studied from both data-driven and model-based approaches, where many learning-based trackers [2], [3] have achieved state-of-the-art performance on widely used benchmarks such as the nuScenes dataset [4]. From a system-level perspective, designing all submodules—such as the object detector, MOT, and guidance, navigation, and control (GNC) system—based on neural networks may impose a significant computational burden on the entire system and raise concerns regarding guaranteed reliability. Therefore, a model-based approach remains meaningful at the system-level and provides mathematically interpretable bases, and these strengths have driven the development of several well-established theoretical base and algorithms for MOT [5], [6], [7], [8].

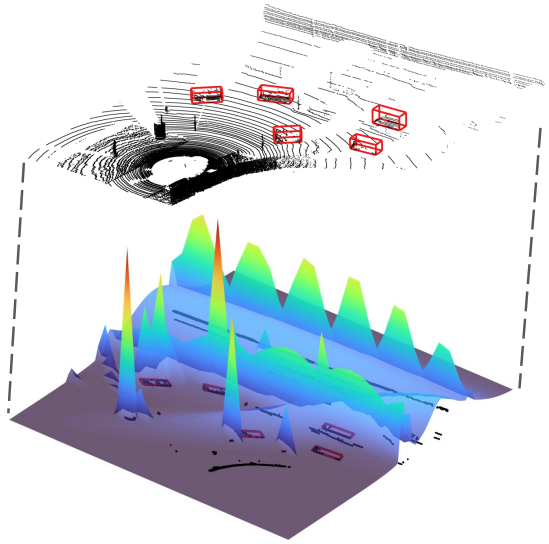


Fig. 1: Gaussian mixture modeling of the birth intensity function, where red Bboxes indicate true object annotations.

Traditionally, model-based trackers have been designed using the Kalman filter (KF) for linear systems and its variants, as well as non-parametric filters for nonlinear and non-Gaussian systems. These filters are often combined with various data association methods, ranging from the nearest neighbor (NN) to global NN (GNN), joint probabilistic data association (JPDA), and multiple hypothesis tracking (MHT) [9]. A key limitation of these approaches is that they do not explicitly model the number of objects or mathematically interpret the uncertainty of undetected, appearing, and disappearing objects within the Bayesian framework, as they rely on rule-based or heuristic designs. Random finite set (RFS) theory [10], [11], [12] provides a rigorous foundation to address these challenges, leading to the development of the Poisson-based probability hypothesis density (PHD) filters [5], [13] and the δ-generalized labeled multi-Bernoulli

*This work was supported in part by the Unmanned Vehicles Core Technology Research and Development Program through the National Research Foundation of Korea (NRF), Unmanned Vehicle Advanced Research Center (UVARC) funded by the Ministry of Science and ICT, the Republic of Korea (No. 2020M3C1C1A01086408), and in part by Theater Defense Research Center funded by Defense Acquisition Program Administration under Grant UD240002SD.

[1]Hanyeol Lee, Taeyoon Kim, and Chan Gook Park are with the Department of Aerospace Engineering, and Automation and Systems Research Institute (ASRI), Seoul National University, Seoul 08826, South Korea. han2110@snu.ac.kr; tykim512@snu.ac.kr; chanpark@snu.ac.kr

[2]Yeongkwon Choe is with the Department of Mechatronics Engineering, Kangwon National University, Chuncheon 24341, South Korea. ychoe@kangwon.ac.kr

(δ-GLMB) [6] and Poisson multi-Bernoulli (PMB) and PMB mixture (PMBM) filters [14], [15] within the multi-Bernoulli family.

In the field of LiDAR object tracking, PMB and PMBM filters, whose structures are composed of Poisson and multi-Bernoulli components, have demonstrated their effectiveness in real-world MOT scenarios. Since RFS-M3 [16], which first introduced an RFS-based formulation into this domain, GNN-PMB [17], which designs hypothesis management under a GNN scheme, and OptiPMB [18], which incorporates measurement-driven birth models and adaptive detector designs, have achieved superior performance on the nuScenes benchmark [4]. These studies confirm that RFS-based approaches are well established in LiDAR MOT, while significant potential remains for further performance enhancement.

We aim to leverage cues extracted from raw point clouds to provide additional information to the estimator. The distribution of point clouds offers spatial cues regarding where objects may be born, which plays an important role in reducing false negatives, i.e., objects that exist but are not tracked, thereby supporting the safe design of autonomous systems. Moreover, unlike previous studies that rely solely on spatial probabilistic distances, we exploit cues of object shape by observing object point clouds in more detail. The differences in the extracted shape descriptions are mapped into the probabilistic domain and jointly utilized with spatial probabilistic distances for hypothesis determination.

In this paper, we propose a PMB filter-based MOT algorithm, namely MOTCues, in which cues extracted from raw point clouds are formulated and integrated into corresponding components of the filter. Specifically, the object birth prior in the Poisson part is formulated with Gaussian mixture model (GMM) fitting within the point cloud as shown in Fig. 1. In addition, the local spatial distribution of object point clouds is interpreted by constructing a bounding box (Bbox)-centric descriptor, enabling object shape information to be incorporated into track–measurement association. The main contributions are as follows.

- We propose an RFS-based MOT algorithm that utilizing raw point cloud cues, including the birth prior for effective initialization of new-born objects and shape information for precise hypothesis management.
- A novel birth prior model is formulated through GMM fitting to consistently remain within the conjugate family of the PMB filter, while the shape information is integrated using point cloud descriptors and probabilistic mapping in a simple yet effective manner.
- The effectiveness of the proposed modules is demonstrated on the KITTI object tracking dataset [19] using traditional MOT metrics, and superior performance is achieved compared to baseline model-based MOT algorithms on the nuScenes benchmark [4].

II. Related works

In this section, we review data-driven and model-based MOT methods for LiDAR systems under the track-by-detection approach.

A. Data-driven MOT

SimTrack [20] jointly solves detection and tracking by proposing a matching network model formulated in the time domain, similar to the track-before-detect approach [21]. OGR3MOT [22] also addresses joint association between tracks and detections in a data-driven approach by designing a learnable neural message-passing network. Following the recent trend of neural network architectures, 3DMOT-Former [23] implements a tracking algorithm with a transformer architecture, including an online training strategy to adapt to the distribution shift between training and inference. ShaSTA [2] introduces a network that exploits an affinity matrix to capture an object's shape and the spatio-temporal correlation between detections and tracks. Meanwhile, several belief-propagation (BP)-based studies [24], [25] enhance the completeness of probabilistic frameworks by incorporating components processed through neural networks into the BP process. Although data-driven approaches achieve high accuracy and demonstrate considerable potential, their reliance on neural networks imposes additional computational demands, which may burden the overall system's limited capacity.

B. Model-based MOT

1) Traditional filter: Model-based MOT originated in the aerospace field and has progressively expanded into diverse domains such as defense, autonomous vehicles [26], and robotics [27], proving its role as an essential building block across various systems [28]. Traditionally, KF and their variants, as well as nonparametric filters and the interacting multiple model (IMM) framework [8] for handling unknown motion models, have been widely employed. These estimators have often been combined with data association methods, including JPDA [7] and MHT [9], which have remained among the most popular forms.

Building on these earlier approaches, LiDAR-based tracking has also adopted random-vector models. AB3DMOT [29] demonstrated the applicability of traditional models to the LiDAR domain by employing KF as the state estimator and the Hungarian algorithm for data association. Probabilistic MOT [30] introduced probabilistic distance metrics by leveraging the Mahalanobis distance for association. SimpleTrack [31] formulated widely used techniques such as non-maximum suppression (NMS) and generalized IoU (GIoU) to enhance the performance of modern MOT algorithms, providing valuable analyses as well. ACKF [32] addressed data association with an affinity model considering appearance, geometry, and distance in an adaptive cubature KF. However, these studies, which formulate states in the random-vector space, fail to capture the spatial uncertainty of undetected targets as well as the uncertainty in the number of objects.

2) Random finite set: RFS theory, introduced by Mahler, provides a powerful framework that elegantly formulates MOT within the Bayesian paradigm [10], [11], [12]. An RFS is defined as a set of random vectors with finite and unique characteristics. Several probabilistic representations

have been proposed, such as finite set statistics (FISST), probability generating functional (PGFL), and multi-object probability density function (PDF). Among these, the multi-object PDF offers advantages due to its intuitive nature and ease of interpretation.

A multi-object PDF consists of a distribution over the cardinality (the number of objects) and a PDF for the multi-object state given the cardinality. The PHD filter, one of the early RFS-based filter applies a first-moment approximation [5], [33]. In the point process field, the PHD is also referred to as the intensity function, representing the expected number of targets per unit hyper-volume. This novel concept led to numerous extensions, including the CPHD filter [13], which generalizes the PHD by incorporating the distribution of the number of targets into the Bayesian framework. Subsequently, filters within the multi-target multi-Bernoulli (MeMBer) family, such as the δ-GLMB filter [6] and the PMB and PMBM filters for LiDAR systems [14], [15], were developed to ensure stable tracking performance.

3) RFS-based approaches in LiDAR MOT: RFS-M3 [16] was the first to formulate an RFS-based method for LiDAR MOT by constructing a PMBM filter with multiple measurement models, including both point and Bbox inputs. GNN-PMB [17] applies GNN to hypothesis management to establish a PMB-based algorithm and provides a detailed analysis of various parameters to ensure reliable performance under diverse conditions. OptiPMB [18] was proposed in the PMB framework, incorporating a birth model derived from measurements and an adaptive detection scheme. We integrate informative cues obtained from point clouds with more detailed observation into the PMB filter in a probabilistic form. This integration achieves notable reductions in ID switches, leading to a design better suited for the safe operation of autonomous systems, while preserving the conjugate family of the PMB filter.

III. Tracker overview with basis of PMB filter

A. State and measurement set definition

The objects are estimated in the bird's-eye view (BEV) domain, and accordingly the state defined as an RFS is expressed as

$$X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\} \in \mathcal{F}(\mathbb{R}^{n_x}), \tag{1}$$

where

$$\boldsymbol{x}_i = [\, p_x \; p_y \; v_x \; v_y \,]^T. \tag{2}$$

X is a set of single-target vectors, where $\mathcal{F}(\cdot)$ denotes the class of all finite subsets of $\mathbb{R}^{n_x}$, the n_x-dimensional real vector space. p_x and p_y denote the 2D object position, and v_x and v_y represent the 2D object velocity. Similarly, a multi-object measurement is defined as

$$Z = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_n\} \in \mathcal{F}(\mathbb{R}^{n_z}), \tag{3}$$

where

$$\boldsymbol{z}_i = [\, p_x \; p_y \,]^T. \tag{4}$$

All states and measurements are constructed in a common frame with the known ego-vehicle pose.

B. Poisson multi-Bernoulli (PMB) model

The RFS of the PMB filter consists of two types of modeling assumptions: a Poisson RFS for undetected objects, X_{p}, and a multi-Bernoulli RFS for potentially detected objects, X_{mb} [15]. With the independence assumption of each component, the density is expressed as

$$f(X) = \sum_{X_{\mathrm{p}} \uplus X_{\mathrm{mb}} = X} f^{\mathrm{p}}(X_{\mathrm{p}})\; f^{\mathrm{mb}}(X_{\mathrm{mb}}). \tag{5}$$

$\uplus$ denotes the disjoint union, $f^{\mathrm{p}}(X_{\mathrm{p}})$ is the density of the Poisson RFS, and $f^{\mathrm{mb}}(X_{\mathrm{mb}})$ is the density of the multi-Bernoulli RFS. The Poisson part is expressed as

$$f^{\mathrm{p}}(X_{\mathrm{p}}) = e^{-\int \lambda(\boldsymbol{x})\, d\boldsymbol{x}} [\lambda(\boldsymbol{x})]^{X_{\mathrm{p}}}. \tag{6}$$

$\lambda(\cdot)$ is the intensity function, and the multi-Bernoulli part is expressed as

$$f^{\mathrm{mb}}(X_{\mathrm{mb}}) = \sum_{X_1 \uplus \ldots \uplus X_n = X_{\mathrm{mb}}} \prod_{i=1}^{n} f^{\mathrm{Ber}}(X_i). \tag{7}$$

Each X_i denotes the RFS of a potentially detected object, and Bernoulli densities are defined as

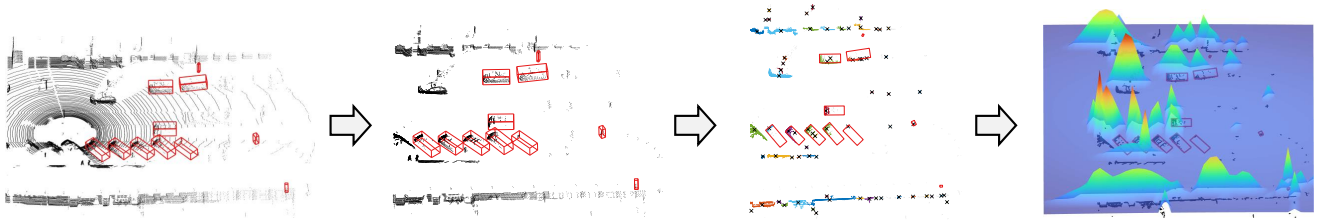
$$f^{\mathrm{Ber}}(X_i) = \begin{cases} 1 - r_i, & X_i = \emptyset, \\ r_i\, p_i(\boldsymbol{x}), & X_i = \{\boldsymbol{x}\}, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

r_i is the existence probability and $p_i(\boldsymbol{x})$ is the state density conditioned on existence.

C. The object tracker overview

As described in the previous subsection, the state and measurement of our tracker are represented as a two-dimensional RFS model, which follows the definition adopted in most existing LiDAR trackers. Since the framework is also based on the track-by-detection approach, the presence of an independent detector is assumed. The intensity of undetected objects is probabilistically formulated using a Poisson point process (PPP), under the assumption of a Poisson RFS introduced in Section III-B. This component consists of objects that are hypothesized to exist but remain undetected, as well as newly born objects in the current frame. The component corresponding to undetected objects is propagated from the previous frame while considering the survival probability. The birth intensity function is commonly modeled as a uniform distribution, under the assumption that the probability of object appearance is equally distributed across the region.

We make our first key contribution in the birth modeling part. When prior information on object appearance is available, the birth intensity function can be modeled by incorporating this information. For example, in aerospace applications, regions such as airports or airbases can be modeled with higher intensity. An important consideration is that the family of probability distributions in the PMB filter should remain conjugate in order to avoid the demand for non-parametric implementations, which are computationally expensive. Therefore, we construct the birth intensity function within the same conjugate prior by segmenting the



Raw point cloud | Segmented point cloud | Cluster initialization | Birth intensity function

Fig. 2: Birth intensity function formulation process. From the raw point cloud, points within the desired height are segmented, and the final birth intensity function is obtained through cluster initialization followed by GMM fitting.

desired region in the raw point cloud, initializing clusters, and performing GMM fitting. Further details are provided in section IV-A.

Returning to the overall filtering process, three types of Bayesian updates are performed once measurements are obtained. First, for objects that remain undetected, the Poisson part is propagated while considering the detection probability. Second, when an object is detected for the first time in the current frame, the Poisson prior is updated to a Bernoulli posterior, including the existence probability and the weight of the Gaussian mixture (GM), while accounting for the clutter rate. Third, for already tracked objects, updates are performed with respect to all possible associations between objects and measurements. Finally, among all possible hypotheses, the one with the globally minimum cost is selected using the Murty algorithm [34]. This choice allows extension to PMBM-based formulations when required.

In this part, we present our second key contribution. The cost between tracks and measurements is typically assigned using the Mahalanobis distance, which works well in most cases. However, in complex scenarios where tracks are densely distributed, such errors may cause mismatches and consequently lead to ID switching. To mitigate this issue, we extract object shape information from the raw point cloud within each detected Bbox. We construct a Bbox-centric descriptor, also considering the confidence of each descriptor. Finally, the descriptors are transformed into the probability domain using a sigmoid kernel and are utilized together with the Mahalanobis distance for global hypothesis determination. Further details are provided in section IV-B.

IV. MOTCues object tracker

In this section, we present the mathematical formulation of the overview introduced in Section III. Bayesian filtering is a framework to recursively estimate the probability distribution over the state set. It consists of a prediction step based on prior information such as the motion model and the properties of survival and birth, and an update step based on measurement information including detection results, clutter rate, and detection probability.

A. Prediction of conjugate prior

In the prediction step, the posterior densities of the conjugate prior are propagated through the transition function. This process is expressed by the Chapman–Kolmogorov equation for the multi-object state as

$$f(X_k \,|\, Z_{1:k-1}) = \int g(X_k \,|\, X_{k-1})\, f(X_{k-1} \,|\, Z_{1:k-1})\, \delta X_{k-1}. \tag{9}$$

$g(X_k \mid X_{k-1})$ denotes the multi-object transition density, which characterizes the probabilistic evolution of the state set from time step $k-1$ to k, incorporating the motion model as well as the properties of survival and birth.

1) Undetected objects: Poisson RFS represents undetected objects, which are composed of newborn objects and the posterior from the previous time step as

$$\lambda_{k|k-1}(\boldsymbol{x}) = \lambda^{b}_{k|k-1}(\boldsymbol{x}) + \lambda^{u}_{k|k-1}(\boldsymbol{x}), \tag{10}$$

where

$$\lambda^{u}_{k|k-1}(\boldsymbol{x}) = \int g(\boldsymbol{x} \,|\, \boldsymbol{y})\, p_s(\boldsymbol{y})\, \lambda^{u}_{k-1}(\boldsymbol{y})\, \boldsymbol{dy}. \tag{11}$$

$\lambda^{b}(\boldsymbol{x})$ denotes the birth intensity, $\lambda^{u}(\boldsymbol{y})$ is the undetected intensity, $p_s(\boldsymbol{y})$ is the survival probability.

Birth intensity formulation: Modeling the intensity function with a uniform distribution is simple but does not reflect the information available from detailed observations. Since occluded objects are assumed to be unobservable, the regions where objects exist return point clouds. This observation provides a critical cue for modeling the birth intensity function. Fig. 2 illustrates the proposed birth function formulation process. The main targets of autonomous systems are movable objects such as cars, pedestrians, and bicycles, which are typically observed at low heights. Therefore, by segmenting the point cloud within the desired height range, irrelevant structures such as buildings and ground points can be removed. To preserve the conjugate prior of the PMB filter consistent, the birth function is modeled as a GM distribution [35]. To enable a robust formulation of the birth intensity function, DBSCAN [36] is applied to obtain subsets of the point cloud for initializing the GMM parameters. The birth intensity function is then expressed as

$$\lambda^{b}(\boldsymbol{x}) = p_b \sum_{j=1}^{N_b} \omega^{b}_{j}\, \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}^{b}_{j}, P^{b}_{j}), \tag{12}$$

$$\sum_{j=1}^{N_b} \omega^{b}_{j} = 1, \quad \omega^{b}_{j} \geq 0, \tag{13}$$

where p_b is the birth rate, ω^{b}_{j} is the mixture weight, $\boldsymbol{\mu}^{b}_{j}$ is the mean vector, and P^{b}_{j} is the covariance matrix of the j-th Gaussian component. The parameters $\{\omega^{b}_{j}, \boldsymbol{\mu}^{b}_{j}, P^{b}_{j}\}_{j=1}^{N_b}$ are estimated using the expectation-maximization (EM) framework. Starting from initialized clusters by DBSCAN, the

procedure alternates between computing the responsibilities of samples with respect to each Gaussian component and updating the corresponding parameters. This iterative process maximizes the log-likelihood until convergence.

Assuming that each undetected object shares the same survival probability, the intensity function in (10) is expressed in Gaussian form as

$$\lambda^u(\boldsymbol{x}) = p_s \sum_{j=1}^{N_u} \omega_j^u \mathcal{N}(\boldsymbol{x};\ \Phi\boldsymbol{\mu}_j^u, \Phi P_j^u \Phi^\top + Q). \tag{14}$$

Each Gaussian component from the previous time step, parameterized by $\{\omega_j^u, \boldsymbol{\mu}_j^u, P_j^u\}$, is propagated through the motion model Φ with process noise covariance Q.

2) Potentially detected objects: The prediction of Bernoulli components is divided into two parts: the probability distribution of the state and the existence probability. The i-th state density is expressed in a form similar to the Poisson part as

$$p_{k|k-1}^{(i)}(\boldsymbol{x}) = \int g(\boldsymbol{x}\,|\,\boldsymbol{y})\, p_{k-1}^{(i)}(\boldsymbol{y})\, \boldsymbol{dy}. \tag{15}$$

For the Gaussian system, this becomes

$$p_{k|k-1}^{(i)}(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}\,;\, \Phi\boldsymbol{\mu}_{k-1}^{(i)},\ \Phi P_{k-1}^{(i)} \Phi^T + Q). \tag{16}$$

The existence probability is predicted by considering the i-th survival probability of the object as

$$r_{k|k-1}^{(i)} = p_s^{(i)}\, r_{k-1}^{(i)}. \tag{17}$$

B. Update of conjugate prior

The update of the PMB filter considers three cases: 1) an undetected object that remains undetected, 2) an undetected object that is potentially detected for the first time, and 3) a previously potentially detected object.

1) Undetected objects: When an undetected object remains undetected, the intensity function is updated by considering the probability of missed detection. Specifically, the undetected part is multiplied by $(1 - p_d)$, where p_d denotes the detection probability. The updated intensity function is

$$\lambda_k^u(\boldsymbol{x}) = (1 - p_d)\, \lambda_{k|k-1}(\boldsymbol{x}). \tag{18}$$

2) Potentially detected objects for the first time: When an undetected object is potentially detected for the first time, the corresponding Poisson component is transitioned into a Bernoulli component. For each measurement $\boldsymbol{z}_j \in Z_k$, a new Bernoulli RFS is formed with an existence probability and an associated state density. The detection likelihood term for measurement $\boldsymbol{z}_j$ is defined as

$$e(\boldsymbol{z}_j) = p_d \int p(\boldsymbol{z}_j\,|\,\boldsymbol{x})\, \lambda_{k|k-1}(\boldsymbol{x})\, dx, \tag{19}$$

and the existence probability of the resulting Bernoulli component is given by

$$r_k^{(j)} = e(\boldsymbol{z}_j)/(e(\boldsymbol{z}_j) + c(\boldsymbol{z}_j)). \tag{20}$$

$c(\boldsymbol{z}_j)$ denotes the clutter intensity, which characterizes the spatial distribution of false alarms in the measurement space. The corresponding state density is obtained as

$$p(\boldsymbol{x}\,|\,\boldsymbol{z}_j) = p_d\, p(\boldsymbol{z}_j\,|\,\boldsymbol{x})\, \lambda_{k|k-1}(\boldsymbol{x})/e(\boldsymbol{z}_j). \tag{21}$$

In the Gaussian case, the detection likelihood becomes

$$e(\boldsymbol{z}_j) = p_d \sum_{i=1}^{N_b+N_u} \omega_i\, \mathcal{N}\big(\boldsymbol{z}_j; H\boldsymbol{\mu}_i,\, S_i\big), \tag{22}$$

$$S_i = H P_i H^\top + R. \tag{23}$$

$\boldsymbol{\mu}_i$ and P_i are the mean and covariance of the i-th Gaussian component of the predicted Poisson intensity. The updated state density is then expressed as a GM,

$$p(\boldsymbol{x}\,|\,\boldsymbol{z}_j) = \sum_{i=1}^{N_b+N_u} \tilde{\omega}_i(\boldsymbol{z}_j)\, \mathcal{N}\big(\boldsymbol{x}; \boldsymbol{\mu}_i^+(\boldsymbol{z}_j),\, P_i^+\big), \tag{24}$$

$$\boldsymbol{\mu}_i^+(\boldsymbol{z}_j) = \boldsymbol{\mu}_i + K_i\big(\boldsymbol{z}_j - H\boldsymbol{\mu}_i\big), \tag{25}$$

$$P_i^+ = P_i - K_i S_i K_i^\top. \tag{26}$$

$\tilde{\omega}_i(\boldsymbol{z}_j)$ denotes the normalized weight, K_i is the Kalman gain, and $\boldsymbol{\mu}_i^+(\boldsymbol{z}_j)$ and P_i^+ denote the updated mean and covariance, respectively.

3) Potentially detected objects: For a previously potentially detected object, the Bernoulli component is updated by incorporating the measurements, detection probability, and clutter intensity.

a) Missed detection: The missed-detection hypothesis is constructed, and its corresponding weight is updated as

$$w_k^{(i,0)} = \big(1 - r_{k|k-1}^{(i)} + r_{k|k-1}^{(i)}(1 - p_d)\big)\, w_{k-1}^{(i)}, \tag{27}$$

and the existence probability is updated as

$$r_k^{(i,0)} = r_{k|k-1}^{(i)}(1 - p_d)/(1 - r_{k|k-1}^{(i)} + r_{k|k-1}^{(i)}(1 - p_d)). \tag{28}$$

The state density remains unchanged in this case.

b) Detection: When a measurement $\boldsymbol{z}_j$ is associated with the i-th Bernoulli component, the corresponding detection hypothesis is constructed. The hypothesis weight is updated as

$$w_k^{(i,j)} = r_{k|k-1}^{(i)} p_d \int p(\boldsymbol{z}_j\,|\,\boldsymbol{x})\, p_{k|k-1}^{(i)}(\boldsymbol{x})\, \boldsymbol{dx}\ w_{k-1}^{(i)}, \tag{29}$$

and the state density is updated by incorporating the measurement likelihood as

$$p_k^{(i,j)}(\boldsymbol{x}) = \frac{p_d\, p(\boldsymbol{z}_j\,|\,\boldsymbol{x})\, p_{k|k-1}^{(i)}(\boldsymbol{x})}{\int p_d\, p(\boldsymbol{z}_j\,|\,\boldsymbol{x})\, p_{k|k-1}^{(i)}(\boldsymbol{x})\, \boldsymbol{dx}}. \tag{30}$$

In this case, the existence probability is always set to one. In the Gaussian system, the detection likelihood for measurement $\boldsymbol{z}_j$ is

$$e^{(i)}(\boldsymbol{z}_j) = p_d\, \mathcal{N}\big(\boldsymbol{z}_j; H\boldsymbol{\mu}_{k|k-1}^{(i)},\, S_i\big), \tag{31}$$

and the hypothesis weight is updated as

$$w_k^{(i,j)} = e^{(i)}(\boldsymbol{z}_j)\, w_{k-1}^{(i)}. \tag{32}$$

The updated state density is then expressed as

$$p_k^{(i,j)}(\boldsymbol{x}) = \mathcal{N}\big(\boldsymbol{x}; \boldsymbol{\mu}_i^+(\boldsymbol{z}_j),\, P_i^+\big), \tag{33}$$

with the notation consistent with the previous subsection.

TABLE I: The mean GOSPA (↓) results in the KITTI validation set with 100 Monte Carlo trials.

Sequence	002	003	004	005	006	007	008	009	010	011	012	013	014	015
Vanilla PMB	2.601	6.832	5.104	5.945	4.036	1.579	5.536	2.673	7.632	5.181	1.420	2.756	2.872	2.679
+Shape description	2.689	6.903	5.253	6.003	3.957	1.622	5.669	2.707	7.679	5.290	1.549	2.748	3.105	2.727
+Birth function	2.114	4.477	3.907	4.025	2.272	1.383	4.255	2.214	5.375	3.820	1.291	**2.408**	2.475	1.992
Proposed (+Both)	**2.101**	**3.735**	**3.673**	**3.731**	**2.141**	**1.376**	**3.779**	**2.183**	**4.729**	**3.712**	**1.190**	2.414	**2.458**	**1.987**

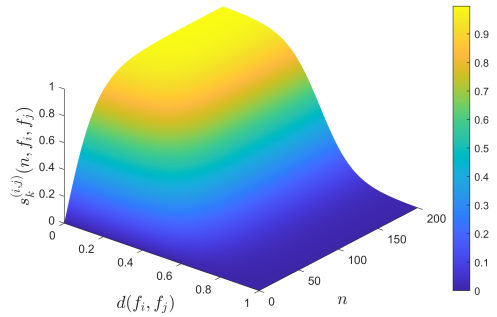


Fig. 3: Visualization of the used kernel with hyperparameters $a = 12$, $b = 0.5$, $a' = 0.05$, $b' = 0$, $c' = 2$, and $d' = -1$.

Object shape descriptor: For the global determination of hypotheses, the weights in (32) are employed, which only reflect probabilistic distances in the state space. To achieve more accurate and robust tracking, we additionally incorporate information of object shapes by constructing Bbox-centric descriptors from raw point clouds. A descriptor of each Bbox is constructed by computing Fast Point Feature Histograms (FPFH) [37] using the Bbox centroid as the reference point, with only minor additional complexity since the computation is performed in a local region. Using these descriptors, the distance between each track–measurement pair can be calculated and incorporated into the hypothesis weight as a shape similarity.

The reliability of a descriptor depends on the number of points used in its construction: the more points involved, the higher the confidence. To account for this, we propose an object shape measure defined in the space of the chi-square distance between descriptors and the number of points as shown in Fig. 3. The mathematical expression of the sigmoid kernel that maps this measure into the probabilistic domain is given as

$$s_k^{(i,j)}(n, \boldsymbol{f}_i, \boldsymbol{f}_j) = -\frac{\alpha(n)}{1 + \exp\big(a\big(-d(\boldsymbol{f}_i, \boldsymbol{f}_j) + b\big)\big)} + \alpha(n), \tag{34}$$

where

$$\alpha(n) = \frac{c'}{1 + \exp(a'(-n + b'))} + d', \tag{35}$$

$$d(\boldsymbol{f}_i, \boldsymbol{f}_j) = \chi^2(\boldsymbol{f}_i, \boldsymbol{f}_j). \tag{36}$$

$s_k^{(i,j)}$ denotes the matched shape probability, $\alpha(n)$ represents the confidence depending on the number of points n, and $d(\boldsymbol{f}_i, \boldsymbol{f}_j)$ is the chi-square distance between the descriptors, where $\boldsymbol{f}_i$ and $\boldsymbol{f}_j$ denote the object descriptors. All other parameters are hyperparameters.

The resulting $s_k^{(i,j)}$ is combined with the hypothesis weight in (32) through a simple weighted sum to form the final hypothesis weight $h_k^{(i,j)}$ as

$$h_k^{(i,j)} = \beta\, w_k^{(i,j)} + (1 - \beta)\, s_k^{(i,j)}. \tag{37}$$

$\beta \in [0, 1]$ is a weighting parameter. Based on the set of hypothesis weights, a cost matrix is constructed by taking the negative logarithm of the hypothesis weights, which converts the problem into a global assignment optimization. The optimal associations are then obtained by solving this assignment problem using Murty algorithm [34].

V. Evaluations

We validate the proposed method using the KITTI object tracking dataset [19] and the nuScenes dataset [4], both of which are major benchmarks for autonomous driving due to their large scale and high-quality data. The KITTI dataset is vision-centric, and its benchmark metrics are evaluated by projecting tracking results onto the 2D image plane. To evaluate more directly, we assess object positions against ground truth using the generalized optimal sub-pattern assignment (GOSPA) [38], with parameters $p = 2$, $c = 8$, and $\alpha = 2$, which is a primary metric in the target tracking field. Through this evaluation, we verify the effectiveness of the proposed module compared to the vanilla PMB filter [15] as a baseline.

The nuScenes dataset is one of the most widely used benchmarks for MOT in autonomous driving and provides multiple evaluation metrics, including average multi-object tracking accuracy (AMOTA), average multi-object tracking precision (AMOTP), and identity switches (IDS). Among these, AMOTA serves as the most comprehensive performance indicator and is widely adopted in the MOT field.

A. KITTI object tracking dataset

This subsection aims to verify the proposed modules by constructing tracking scenarios based on real-world data. Since we adopt a track-by-detection approach, detection results are generated by applying white Gaussian noise with a standard deviation of 0.3 m to the ground-truth object positions with a detection probability of 0.9 in sequences 2–15 of the validation set. For comparison, we employ a PMB filter without the proposed birth prior and shape descriptor, which we denote as the Vanilla PMB. In addition, we report results for each module applied individually, while the case using both modules together is denoted as Proposed. The results of 100 Monte Carlo trials are summarized in Table I.

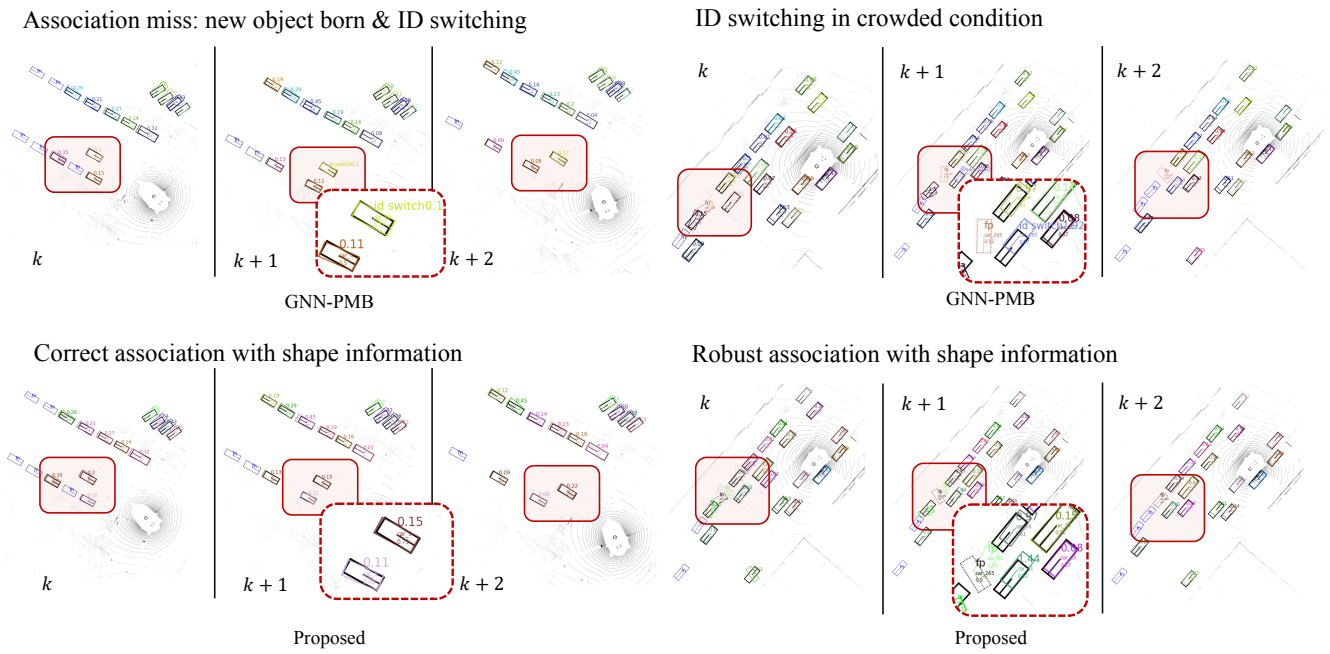


Fig. 4: Quantitative comparison between the proposed method and GNN-PMB [17] in nuScenes validation set.

TABLE II: Detailed benchmark results on the nuScenes validation set.

	AMOTA↑								AMOTP↓	IDS↓
	Bicycle	Bus	Car	Motorcycle	Pedestrian	Trailer	Truck	Mean		
GNN-PMB [17]	0.513	0.854	**0.849**	0.723	0.807	0.506	**0.695**	0.707	0.560	650
Proposed	**0.532**	**0.857**	0.848	**0.737**	**0.814**	**0.510**	**0.695**	**0.713**	**0.555**	**564**

TABLE III: Benchmark results on nuScenes validation set

	Methods	AMOTA↑	AMOTP↓
Learning base	CenterPoint [39]	0.665	0.567
	OGR3MOT [22]	0.693	0.627
	SimpleTrack(2Hz) [31]	0.687	0.573
	SimpleTrack(10Hz) [31]	0.696	0.547
	NEBP [25]	0.708	-
	ShaSTA [2]	**0.728**	**0.544**
Model base	AB3DMOT [29]	0.578	0.807
	GNN-PMB [17]	0.707	0.560
	Proposed	**0.713**	**0.555**

Overall, the birth prior module leads to performance improvements, and the combination of both modules yields the lowest error, indicating the presence of a synergistic effect. The synergy stems from the complementary roles of the two modules: the expansively defined birth prior alleviates false negatives, which can be fatal in autonomous systems, and the shape description enhances robustness in data association, reducing mismatches.

B. NuScenes Tracking Dataset

In this subsection, our primary baseline is the GNN-PMB [17] filter. Our algorithm for verification on the nuScenes benchmark is implemented based on the publicly available GNN-PMB open-source code[1], with our contributions. Therefore, performance differences with GNN-PMB directly indicate the effectiveness of the proposed modules. For fair comparison, the common object detector CenterPoint [39] is used to generate detection results across all experiments.

[1]https://github.com/chisyliu/GnnPmbTracker

TABLE IV: Benchmark results with previously published RFS-based MOT algorithms on the nuScenes test set.

	AMOTA↑	AMOTP↓	IDS↓
RFS-M3 (IEEE ICRA 2021) [16]	0.619	0.752	1525
GNN-PMB (IEEE TIV 2023) [17]	0.678	**0.560**	770
Proposed	**0.679**	0.568	**671**

Table III summarizes the overall performance evaluation against other published algorithms on the validation set. The results demonstrate that our model achieves high performance among model-based approaches, showing notable improvements over GNN-PMB in particular. Table II presents detailed benchmark results, where improvements are observed across most categories, with a remarkable reduction in IDS, evidencing the effectiveness of the proposed shape descriptor. Finally, the nuScenes test set results are presented in Table IV, where our method attains improved AMOTA and consistently fewer IDS than GNN-PMB, following the same tendency observed in the validation set. Fig. 4 presents a case study comparing GNN-PMB with our algorithm. The results show that the proposed shape description module enables accurate association in typical scenarios and facilitates robust tracking in crowded environments.

VI. Conclusion

In this work, we proposed MOTCues, an object tracker grounded in RFS theory. Informative cues extracted from detailed point cloud observations were incorporated into the PMB filter through a birth intensity function for accurate initialization of new-born objects and a Bbox-centric descriptor for robust hypothesis association. The proposed modules were validated on the KITTI dataset, and superior performance over the baseline was demonstrated on the nuScenes benchmark. We plan to integrate the proposed tracker with simultaneous localization and mapping algorithms, advancing toward the broader objectives of autonomous systems.

References

[1] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, and T. K. Kim, "Multiple object tracking: A literature review," *Artificial Intelligence*, vol. 293, Art. no. 103448, 2021.

[2] T. Sadjadpour, J. Li, R. Ambrus, and J. Bohg, "ShaSTA: Modeling Shape and Spatio-Temporal Affinities for 3D Multi-Object Tracking," *IEEE Robotics and Automation Letters*, vol. 9, no. 5, pp. 4273-4280, 2024.

[3] X. Li, D. Liu, Y. Wu, X. Wu, L. Zhao and J. Gao, "Fast-Poly: A Fast Polyhedral Algorithm for 3D Multi-Object Tracking," *IEEE Robotics and Automation Letters*, vol. 9, no. 11, pp. 10519-10526, 2024.

[4] 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," *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11621-11631.

[5] R. Mahler, "Multitarget Bayes Filtering via First-Order Multitarget Moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152-1178, 2004.

[6] B. T. Vo and B. -N. Vo, "Labeled Random Finite Sets and Multi-Object Conjugate Priors," *IEEE Transactions on Signal Processing*, vol. 61, no. 13, pp. 3460-3475, 2013.

[7] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion*, Storrs CT, USA: YBS Publishing Storrs, 2011

[8] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting Multiple Model Methods in Target Tracking: A Survey," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 1, pp. 103-123, 2002.

[9] S. S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Norwood, MA: Artech, 1999.

[10] R. Mahler, *Advances in Statistical Multisource-Multitarget Information Fusion*, Norwood, MA, USA: Artech House, 2014.

[11] M. Mallick, V. Krishnamurthy, and B.-N. Vo, *Integrated Tracking, Classification, and Sensor Management: Theory and Applications*, Hoboken, NJ, USA: Wiley, 2012.

[12] B. -N. Vo, B. T. Vo, T. T. Nguyen, and C Shim, "An Overview of Multi-Object Estimation via Labeled Random Finite Set," *IEEE Transactions on Signal Processing*, vol. 72, pp. 4888-4917, 2024.

[13] R. Mahler, "PHD Filters of Higher Order in Target Number," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 4, pp. 1523-1543, 2008.

[14] J. L. Williams, "Marginal Multi-Bernoulli Filters: RFS Derivation of MHT, JIPDA, and Association-Based MeMBer," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 3, pp. 1664-1687, 2015.

[15] Á. F. García-Fernández, J. L. Williams, K. Granström, and L. Svensson, "Poisson Multi-Bernoulli Mixture Filter: Direct Derivation and Implementation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 4, pp. 1883-1901, 2018.

[16] S. Pang, D. Morris, and H. Radha, "3D Multi-Object Tracking using Random Finite Set-based Multiple Measurement Models Filtering (RFS-M3) for Autonomous Vehicles," *in Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 13701-13707.

[17] J. Liu, L. Bai, Y. Xia, T. Huang, B. Zhu, and Q. L. Han, "GNN-PMB: A Simple but Effective Online 3D Multi-Object Tracker Without Bells and Whistles," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 2, pp. 1176-1189, 2022.

[18] G. Ding et al., "OptiPMB: Enhancing 3D Multi-Object Tracking with Optimized Poisson Multi-Bernoulli Filtering," arXiv preprint, 2025. [Online]. Available: https://arxiv.org/abs/2503.12968.

[19] A. Geiger, P. Lenz and R. Urtasun, "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite," *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 3354-3361.

[20] C. Luo, X. Yang, and A. Yuille, "Exploring Simple 3D Multi-Object Tracking for Autonomous Driving," *in Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 10488-10497.

[21] S. M. Tonissen and Y. Bar-Shalom, "Maximum Likelihood Track-Before-Detect With Fluctuating Target Amplitude," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 3, pp. 796-809, 1998.

[22] J. N. Zaech, A. Liniger, D. Dai, M. Danelljan, and L. V. Gool, "Learnable Online Graph Representations for 3D Multi-Object Tracking," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5103-5110, 2022.

[23] S. Ding, E. Rehder, L. Schneider, M. Cordts, and J Gall, "3DMOTFormer: Graph Transformer for Online 3D Multi-Object Tracking," *in Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 9784-9794.

[24] S. Wei, M. Liang, and F. Meyer, "A New Architecture for Neural Enhanced Multiobject Tracking," *in Proceedings of International Conference on Information Fusion (FUSION)*, 2024.

[25] M. Liang and F. Meyer, "Neural enhanced belief propagation for multiobject tracking," *IEEE Transactions on Signal Processing*, vol. 72, pp. 15-30, 2023.

[26] Z. Zhu, J. Zhao, K. Huang, X. Tian, J. Lin, and C. Ye, "LIMOT: A Tightly-Coupled System for LiDAR-Inertial Odometry and Multi-Object Tracking," *IEEE Robotics and Automation Letters*, vol. 9, no. 7, pp. 6600-6607, 2024.

[27] K. M. Judd and J. D. Gammell, "Multimotion visual odometry," *The International Journal of Robotics Research*, vol. 43, no. 8, pp. 1250-1278, 2024.

[28] B.-N. Vo et al., *Multitarget Tracking* in Wiley Encyclopedia of Electrical and Electronics Engineering. New York, NY, USA: Wiley, 2015.

[29] X. Weng, J. Wang, D. Held, and K. Kitani, "3D Multi-Object Tracking: A Baseline and New Evaluation Metrics," *in Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 10359-10366.

[30] H. Chiu, J. Li, R. Ambruș, and J. Bohg, "Probabilistic 3D Multi-Modal, Multi-Object Tracking for Autonomous Driving," *in Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 14227-14233.

[31] Z. Pang, Z. Li, and N. Wang, "Simpletrack: Understanding and rethinking 3d multi-object tracking," *in Proceedings of European Conference on Computer Vision (ECCV)*, 2022, pp. 680-696.

[32] G. Guo and S. Zhao, "3D Multi-Object Tracking With Adaptive Cubature Kalman Filter for Autonomous Driving," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 1, pp. 512-519, 2022.

[33] B. -N. Vo and W. K. Ma, "The Gaussian Mixture Probability Hypothesis Density Filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091-4104, 2006.

[34] K. G. Murty, "An Algorithm for Ranking all the Assignments in Order of Increasing Cost" *Operations research*, vol. 16, no. 3, pp. 682-687, 1968.

[35] C. E. Rasmussen, "The Infinite Gaussian Mixture Model," *in Proceedings of Advances in neural information processing (NIPS) 12*, 1999.

[36] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN," *ACM Transactions on Database Systems*, vol. 42, no. 3, pp. 1-21, 2017.

[37] R. B. Rusu, N. Blodow, and M. Beetz, "Fast Point Feature Histograms (FPFH) for 3D registration," *in Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 3212-3217.

[38] A. S. Rahmathullah, Á. F. García-Fernández and L. Svensson, "Generalized optimal sub-pattern assignment metric," *in Proceedings of International Conference on Information Fusion (FUSION)*, 2017.

[39] T. Yin, X. Zhou, and P. Krahenbuhl, "Center-Based 3D Object Detection and Tracking," *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 11784-11793.