Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Self-Supervised Depth Correction via Temporal 3D Consistency under Ego-Motion

Falak Niaz
Kookmin University
JaeJun Yoo
Electronics and Telecommunications Research Institute
Yeong Min Jang
yjang@kookmin.ac.kr

Kookmin University

Article

Keywords:

Posted Date: September 3rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10380476/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Self-Supervised Depth Correction via Temporal 3D Consistency under Ego-Motion

Falak Niaz[1], **JaeJun Yoo**[2], **and Yeong Min Jang**[1,*]

[1]Department of Electronics Engineering, Kookmin University, Jeongneung-ro 77, Seoul, 02707, Seongbuk-gu, South Korea
[2]DNA+ Drone Platform Research Center, Electronics and Telecommunications Research Institute, Daejeon, South Korea
[*]Corresponding Author: Yeong Min Jang; yjang@kookmin.ac.kr

ABSTRACT

Accurate and temporally stable three-dimensional (3D) localization of surrounding vehicles is critical for autonomous driving, particularly in complex urban and highway enviroments where dynamic traffic and varying object densities are present. Although camera-light detection and ranging (LiDAR) fusion provides complementary semantic and geometric cues, object depth estimation remains sensitive to LiDAR sparsity and frame-wise noise, especially at longer ranges or in high-speed scenarios. Conventional pipelines associate two-dimensional (2D) detection with per-frame LiDAR depth, neglecting temporal coherence and ego-motion, which leads to inconsistent 3D trajectories. In addition, most learning-based methods depend on costly 3D annotations, limiting scalability. This paper proposes an object-level, uncertainty-aware self-supervised refinement framework that learns adaptive 3D corrections under ego-motion using temporal geometric consistency, without requiring 3D bounding-box supervision. Rather than estimating dense depth, the method focuses on object-level refinement. A lightweight network predicted per-object depth corrections using image features and LiDAR statistics, including aggregated depth descriptors such as mean, median, variance, minimum or maximum depth, and point density computed within each object-associated LiDAR cluster. Training was performed without 3D ground truth by enforcing trajectory consistency in a global coordinate frame, with explicit ego-motion compensation to separate apparent and true motion. This formulation leveraged motion-induced geometric constraints as intrinsic supervision, enabling learning directly from raw driving sequences obtained from real-world benchmark datasets. The proposed method reduced mean CV error by 9.40%, median CV error by 14.45%, and jerk mean by 21.45%. Robustness analyses under varying LiDAR sparsity further confirmed stable performance in both dense urban traffic and high-speed highway conditions. The results indicated that ego-motion-aware temporal consistency provides an effective self-supervised signal for improving object-level 3D localization stability without requiring annotated 3D bounding boxes.

1 Introduction

Perception of three-dimensional (3D) dynamic road environments in real time is fundamental to autonomous driving and advanced driver assistance systems. Accurate localization and motion estimation of surrounding vehicles enable safe navigation, effective planning, and collision avoidance[1]. Accordingly, contemporary perception systems employ multi-sensor fusion, integrating cameras and light detection and ranging (LiDAR) to Combine semantic richness from images with geometric precision from sparse point clouds. This approach is widely utilized for object detection, semantic segmentation, and scene understanding, and continues to advance to meet practical deployment challenges[2]. Although supervised fusion models have achieved strong performance in 3D perception, they typically require large volumes of annotated 3D bounding boxes or dense depth maps, which are costly and time-intensive to obtain at scale[3]. Moreover, many fusion pipelines operate on a frame-wise basis, associating two-dimensional (2D) detections with instantaneous LiDAR measurements without modeling temporal consistency. This approach is susceptible to noisy depth estimates, particularly under sparse, distant, or occluded LiDAR returns, resulting in unstable object trajectories and inconsistent motion estimation. These limitations are amplified at long ranges and in challenging environmental conditions critical to autonomous navigation[4].

Self-supervised learning has emerged as an effective alternative to supervised methods by exploiting geometric and temporal constraints to eliminate the need for manual annotations[5]. In monocular video-based settings, photometric and multi-view consistency are used to jointly estimate depth and camera motion[6]. Extending these principles to multi-sensor fusion has shown promise in tasks such as depth completion and long-range 3D perception, where temporal cues and sparse LiDAR enhance representation learning. Moreover, recent advances in multi-frame and multi-camera self-supervised depth estimation demonstrate that temporal context significantly improves geometric reconstruction and pose estimation[7].

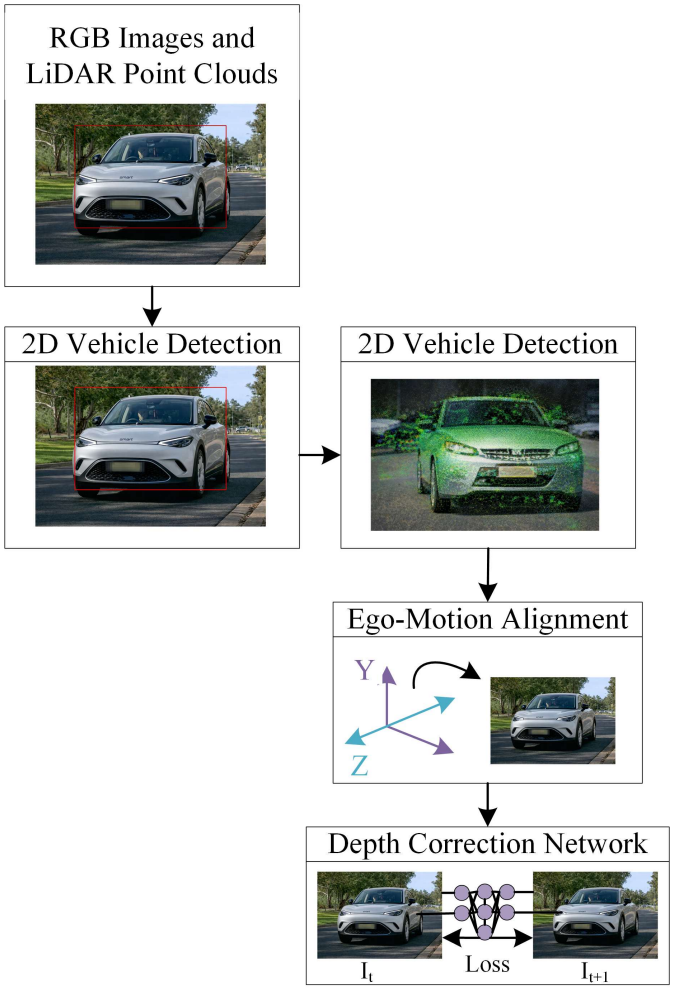


Figure 1. Self-Supervised depth correction pipeline for vehicle detection and depth refinement using RGB images and LiDAR point clouds.

However, most existing approaches emphasize dense depth prediction or feature alignment, and do not directly address stable object-level 3D localization under ego-motion without relying on 3D ground truth. In contrast to prior frameworks that rely on explicit annotations or underutilize temporal information, this work proposes a self-supervised temporal depth correction framework for object-level vehicle localization. Instead of learning dense scene depth or using labeled 3D bounding boxes, the method refines object-level depth through a lightweight correction network trained via temporal motion consistency, with explicit integration of ego-motion from Inertial Measurement Unit (IMU) data. By modeling object trajectories in a global coordinate frame and enforcing a constant-velocity (CV) constraint, the network corrects LiDAR-initialized depth to produce stable and smooth 3D trajectories. This formulation leverages motion-induced geometric constraints as intrinsic supervision, enabling training directly from Raw driving sequences without manual labels.

In addition to accuracy, computational efficiency and real-time feasibility are critical requirements for deployment in autonomous driving systems. The proposed framework is designed with a lightweight architecture that operates on object-level features rather than dense depth maps, significantly reducing computational overhead. By leveraging sparse LiDAR statistics and compact image representations, the method avoids expensive dense 3D processing and large-scale network architectures. This design enables efficient inference suitable for real-time or near real-time operation on standard automotive-grade hardware, making it practical for integration into real-world perception pipelines.

The key contributions of this paper are summarized as follows:

i. A self-supervised framework for object-level depth correction based on temporal consistency under ego-motion, eliminating the need for 3D bounding box annotations.

ii. An ego-motion-aware training formulation that decouples object motion from platform motion, enabling globally

consistent trajectory refinement

iii. A lightweight depth refinement network that integrates image features with LiDAR statistics to achieve temporally stable 3D localization.

iv. Extensive evaluation on multiple Karlsruhe Institute of Technology and Toyota Technological Institute at Chicago (KITTI) Raw sequences, including analyses under varying LiDAR sparsity and ablation studies demonstrating the impact of temporal and ego-motion constraints.

In contrast to prior self-supervised methods that emphasize dense depth reconstruction or post-hoc temporal smoothing, the proposed approach formulates object-level depth correction as a learnable function guided by ego-motion-compensated trajectory consistency in a global coordinate frame. The Fig. 1 illustrates a processing pipeline that begins with RGB images and LiDAR point clouds to perform 2D vehicle detection in both modalities. The detected regions are then aligned using ego-motion estimation to ensure temporal and spatial consistency. Finally, a depth correction network refines the scene understanding by minimizing the loss between consecutive frames, improving depth accuracy over time.

The remainder of this manuscript is organized as follows. Section 2 reviews related work and situates the proposed method within camera-LiDAR fusion and self-supervised depth estimation. Section 3 details the proposed ego-motion-aware temporal depth correction framework. Section 4 describes the dataset, preprocessing, training setup, and evaluation metrics. Section 5 presents quantitative and qualitative results, including ablation and robustness analyses. Section 6 concludes the paper and outlines future research directions.

2 Related Work

This section is divided into four subsections. Section Section 2.1 reviews camera-LiDAR fusion methods for 3D perception. Section 2.2 discusses depth completion and sparse LiDAR integration. Section 2.3 examines ego-motion-based self-supervised learning approaches. Section 2.4 addresses temporal modeling strategies for 3D detection and tracking.

2.1 Camera–LiDAR Fusion for 3D Perception

Camera–LiDAR fusion has become a fundamental approach for improving environmental perception in autonomous driving by combining the rich semantic information provided by RGB images with the accurate geometric measurements obtained from LiDAR sensors. Early fusion strategies primarily projected LiDAR points onto the image plane to enrich point cloud representations with semantic information, while later approaches integrated features in bird's-eye-view (BEV) or multi-scale feature spaces to improve 3D object detection accuracy. Recent survey studies have shown that camera–LiDAR fusion remains a central direction in autonomous-driving perception because the two modalities provide complementary semantic and geometric information. Wang et al. reviewed multi-sensor fusion strategies for 3D object detection and categorized fusion methods according to fusion stage, sensor representation, and detection architecture[8]. Such studies highlight that most fusion systems are optimized for supervised 3D detection accuracy rather than temporal object-level localization stability.

Representative methods, including PointPainting[9], BEVFusion[10], and 3D Dual-Fusion[11],[12] have demonstrated substantial improvements in detection performance by jointly exploiting complementary sensor modalities. These methods are generally optimized for supervised 3D object detection and rely on large-scale annotated datasets containing accurate 3D bounding boxes. Despite their excellent detection accuracy, most camera–LiDAR fusion frameworks estimate object depth independently for each frame and do not explicitly enforce temporal consistency across consecutive observations. Consequently, object localization may exhibit frame-to-frame fluctuations caused by sparse LiDAR sampling, measurement noise, and imperfect sensor association.

2.2 Depth Completion and Sparse LiDAR Integration

Depth completion is closely related to the proposed problem because it addresses the reconstruction of dense or reliable depth from sparse LiDAR observations. Uhrig et al[13]. introduced sparsity-invariant convolution for depth completion from sparse laser scans, which became an important foundation for the KITTI depth completion benchmark[13]. More recent surveys have emphasized that sparse LiDAR density remains a major limitation for autonomous-driving depth perception, especially at long ranges and near object boundaries[14]. Recent approaches have incorporated temporal and self-supervised strategies to reduce dependence on dense ground truth. Techniques, such as temporal warping and multi-frame fusion, improve cross-frame depth consistency[15]. while self-supervised methods exploit geometric and temporal constraints as alternative supervision[16]. Learning-based depth completion methods such as DeepLiDAR[17] further demonstrate the benefit of combining sparse LiDAR depth with RGB information for outdoor dense depth prediction[17]. Nevertheless, these approaches generally optimize pixel-level depth reconstruction rather than object-track-level temporal stability.

However, these approaches primarily target dense pixel-wise depth estimation and do not explicitly model object identities or trajectory stability. In contrast, the proposed method refines object-level depth by optimizing 3D trajectory consistency rather than per-pixel reconstruction error. Although these methods substantially improve dense depth reconstruction, they are primarily designed for pixel-level depth estimation rather than object-level localization. Furthermore, they generally optimize photometric reconstruction accuracy instead of temporal trajectory stability, leaving temporal consistency largely unexplored.

2.3 Self-supervised Learning with Ego-Motion

Self-supervised learning has been extensively applied to monocular depth estimation and visual odometry, where geometric constraints serve as supervisory signals without explicit annotations[18,19]. A foundational line of work in self-supervised depth estimation uses view synthesis as the supervisory signal. Zhou et al. jointly learned monocular depth and ego-motion from video without ground-truth depth or pose annotations[19]. Godard et al. later improved self-supervised monocular depth estimation through minimum reprojection, auto-masking, and multi-scale sampling strategies, establishing Monodepth2 as a widely used benchmark method[18]. In multi-sensor settings, visual-LiDAR odometry methods leverage sparse depth information to enhance pose estimation under self-supervision[20]. More recently, multi-frame self-supervised[21] approaches such as ManyDepth exploit sequence information at test time through a cost-volume formulation, showing that temporal context can substantially improve depth estimation[21]. However, these methods remain focused on dense scene depth and do not explicitly refine object-level 3D trajectories under ego-motion.

Recent studies incorporate ego-motion to enforce multi-frame geometric consistency in depth and pose estimation[22,23] demonstrating the effectiveness of motion-based constraints for improving spatial coherence. However, these methods primarily target camera pose estimation or dense depth reconstruction, rather than object-level localization stability. These studies demonstrate that ego-motion provides an effective supervisory signal for self-supervised learning. Nevertheless, existing methods primarily focus on dense depth estimation or visual odometry rather than directly refining object-level 3D positions over time. To further confirms that monocular depth estimation has evolved from supervised dense prediction toward self-supervised and geometry-driven learning, but most methods remain scene-level rather than object-trajectory-level[24].

In contrast, the proposed approach utilizes ego-motion to transform object coordinates into a global reference frame, where a CV prior is imposed on object trajectories. This formulation yields a self-supervised signal specifically designed for vehicle-level 3D localization refinement, rather than scene-level reconstruction, while acknowledging that errors in ego-motion estimation may introduce noise into the transformed trajectories. However, this effect is mitigated by operating over short temporal windows, which limits drift accumulation, and by leveraging object-level consistency constraints that are less sensitive to small global pose inaccuracies. Furthermore, the learning framework is trained to be robust to such noise, enabling the network to compensate for minor ego-motion errors during depth correction.

2.4 Temporal Modeling in 3D Detection and Tracking

Temporal modeling has been applied in 3D detection and tracking through multi-frame aggregation and recurrent architectures to improve robustness and long-range perception[25]. Temporal information has also been widely studied in 3D detection and tracking[26]. CenterPoint formulates 3D objects as centers and extends detection to tracking using object-center association. MPPNet further exploits multi-frame point-cloud sequences through proxy points and hierarchical temporal feature interaction for 3D temporal object detection[27]. In tracking, motion models, such as Kalman filters, are commonly used for post-processing trajectory smoothing. Classical tracking frameworks such as AB3DMOT combine 3D detections with Kalman filtering and data association for efficient multi-object tracking[28]. More recent camera–LiDAR tracking methods, such as CAMO-MOT[29], incorporate both appearance and motion cues to improve 3D multi-object tracking[29]. These methods demonstrate the importance of motion modeling, but their objective is tracking association rather than self-supervised object-depth refinement. However, these approaches do not learn depth correction under self-supervised motion constraints; temporal information is typically applied heuristically or after detection. In particular, temporal aggregation methods primarily combine features or detections across frames (e.g., via concatenation or recurrent fusion) to enhance robustness, but they do not explicitly enforce geometric or motion consistency in the learning process. In contrast, the proposed method embeds temporal consistency directly into the learning objective, where motion coherence across frames serves as an explicit supervisory signal, enabling depth refinement based on motion coherence rather than post-hoc filtering.

Although temporal modeling has significantly improved video-based perception, most existing methods learn temporal feature representations for detection or tracking instead of explicitly refining geometric object depth through self-supervised trajectory consistency. Consequently, the potential of temporal motion constraints for object-level depth correction remains largely unexplored.

2.5 Research Gap and Motivation

Existing research has achieved remarkable progress in camera–LiDAR fusion, self-supervised depth estimation, and temporal perception. However, three important limitations remain.

Table 1
Comparison of existing camera–LiDAR fusion and self-supervised methods. Our method uniquely performs ego-motion-aware, self-supervised object-level depth refinement without requiring 3D bounding-box annotations.

Method	Task	Sensors	3D Sup.	Self-Sup.	Temp.	Ego	Dataset	Output
PointPainting[9]	3D Detection	Cam+LiDAR	✓	✗	✗	✗	KITTI	Semantic painting of LiDAR points.
BEVFusion[10]	3D Detection	Cam+LiDAR	✓	✗	✗	✗	nuScenes	BEV-space feature fusion.
3D Dual-Fusion[11]	3D Detection	Cam+LiDAR	✓	✗	✗	✗	KITTI	Dual-domain feature fusion.
Self-supervised depth completion[16]	Dense Depth	Cam+LiDAR	✗	✓	✓	✗	KITTI	Self-supervised depth with temporal consistency constraint.
Self-VLO[20]	Visual–LiDAR Odometry	Cam+LiDAR	✗	✓	✓	✓	KITTI	Self-supervised visual–LiDAR odometry framework.
Ours	**Vehicle Localization & Trajectory Refinement**	**Cam+LiDAR+IMU**	✗	✓	✓	✓	**KITTI Raw**	**Self-supervised ego-motion-aware object-level depth refinement improving trajectory stability.**

i. First, most camera–LiDAR fusion methods rely on expensive 3D annotations and optimize frame-wise detection accuracy rather than temporal localization stability.

ii. Second, self-supervised depth estimation methods predominantly focus on dense pixel-level reconstruction and rarely exploit temporal consistency for object-level depth refinement.

iii. Third, existing temporal perception methods primarily learn feature representations for detection or tracking, while the geometric consistency of object trajectories is generally treated as a downstream post-processing problem rather than a learning objective.

To address these limitations, this work proposes an ego-motion-aware self-supervised depth refinement framework that directly learns object-level depth corrections using temporal trajectory consistency as supervision. Unlike conventional supervised detection frameworks, the proposed approach requires no annotated 3D bounding boxes and instead exploits physically consistent motion in the global coordinate frame as a supervisory signal.

Table 1 summarizes representative camera-LiDAR fusion and self-supervised methods. Existing approaches predominantly address supervised 3D detection or dense depth completion, often without explicitly exploiting ego-motion for object-level temporal refinement. In contrast, the proposed framework leverages ego-motion-aware temporal consistency within a self-supervised framework to enable object-level depth correction without requiring 3D bounding-box annotations, while maintaining computational efficiency. Although temporal modeling can introduce additional processing overhead, the proposed approach operates on short temporal windows and utilizes compact object-level representations rather than dense spatiotemporal features, thereby limiting the increase in computational cost. Furthermore, the temporal consistency constraint is incorporated primarily during training, ensuring that inference remains lightweight and suitable for real-time deployment.

3 Methodology

This section presents the proposed ego-motion-aware temporal depth correction framework. Section 3.1 provides an overview of the system architecture. Section 3.2 describes the LiDAR-based depth association for object-level localization. Section 3.3 details ego-motion estimation and its transformation into a global coordinate frame. Section 3.4 introduces the self-supervised depth correction network. Section 3.5 defines the temporal 3D consistency loss for label-free training.

3.1 Framework Overview

The proposed framework aims to improve the temporal stability of object-level depth estimates obtained from camera–LiDAR fusion without requiring annotated 3D bounding boxes. Rather than replacing conventional object detection or LiDAR association, the proposed approach refines the initially estimated object depth through self-supervised temporal learning. Figure 2 illustrates the overall framework.

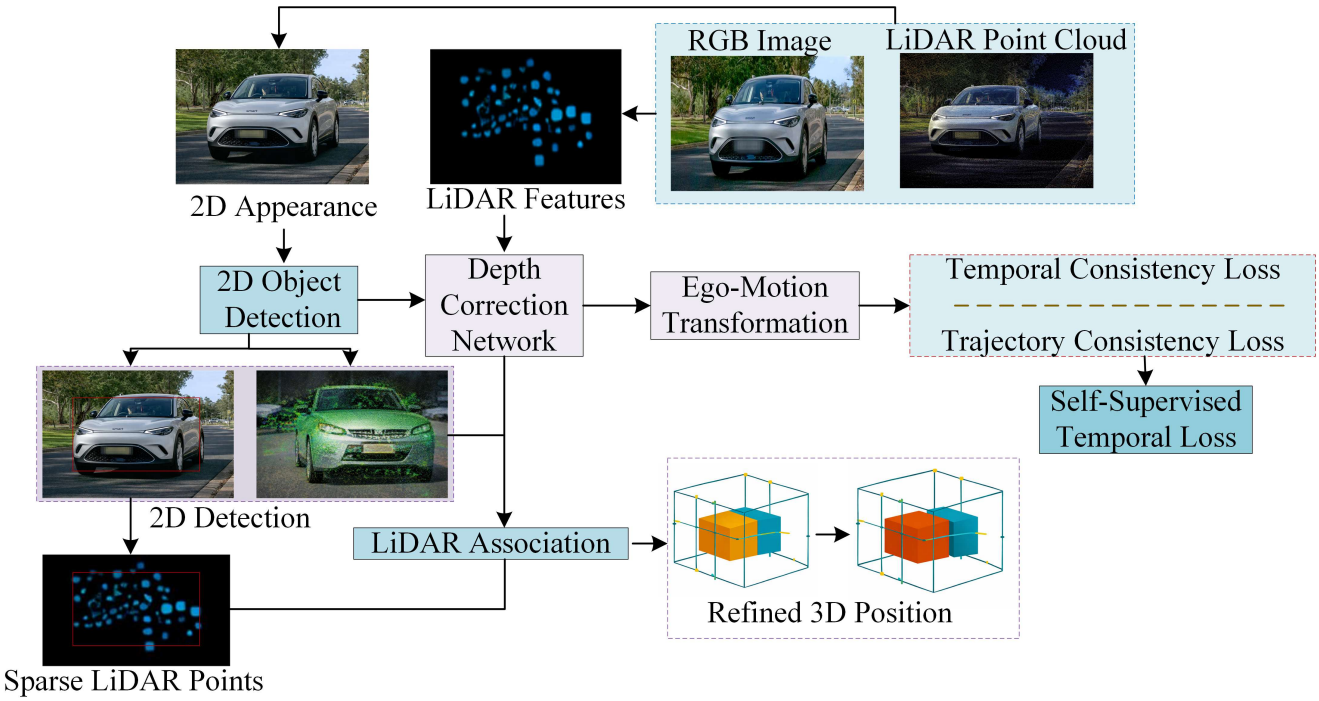


Figure 2. Self-Supervised Temporal Correction Framework

For each image frame, vehicles are first detected in the RGB image using a 2D object detector. Sparse LiDAR points are projected into the image plane and associated with each detected object to obtain an initial depth estimate. These object-level observations are then transformed into a global coordinate frame using ego-motion information derived from GPS/IMU measurements. A lightweight neural network predicts residual depth corrections, while a self-supervised temporal consistency objective enforces physically plausible trajectories across consecutive frames.

Given synchronized RGB images and LiDAR point clouds, the pipeline includes:

i. 2D vehicle detection in image space

ii. LiDAR-based depth association within detection regions

iii. Ego-motion-based coordinate transformation

iv. Self-supervised training using a temporal consistency loss

Let I_t denote the RGB image at time t, and let $\mathscr{P}_t = \{(x_i, y_i, z_i)\}$ represent the corresponding LiDAR point cloud. For each detected object k, an initial depth estimate is derived from LiDAR measurements and subsequently refined through a learned correction function trained using temporal motion constraints in a world coordinate frame, where each object is processed independently based on its associated image region and LiDAR points. In scenes containing multiple vehicles, the framework applies the same refinement process to all detected objects in parallel, enabling scalable handling of crowded environments. Temporal consistency is enforced per object trajectory, ensuring that interactions between objects do not introduce interference in the depth correction process. This design allows the method to maintain computational efficiency while supporting multiple simultaneous object tracks in dense traffic scenarios.

3.2 Object-Level LiDAR Depth Initialization

LiDAR points are projected into the image plane using the calibrated transformation.

$$\mathbf{u}_i = \mathbf{P}\,\mathbf{T}_{\text{velo}\to\text{cam}}\,\mathbf{p}_i, \tag{1}$$

In Eq. (1) $\mathbf{p}_i$ denotes a homogeneous LiDAR point, $\mathbf{T}_{\text{velo}\to\text{cam}}$ is the rigid-body transformation from LiDAR to camera coordinates, and $\mathbf{P}$ is the camera projection matrix.

For each detected bounding box b_t^k, the LiDAR points projected inside the bounding region are collected. The initial object depth z_t^k is estimated as the median of associated depths shown in Eq. (2).

$$z_t^k = \text{median}\left\{ z_i \mid (u_i, v_i) \in b_t^k \right\}. \tag{2}$$

The median depth provides a robust initialization against isolated LiDAR outliers and missing returns. However, because each frame is processed independently, the resulting object depth estimates may exhibit temporal fluctuations caused by measurement sparsity, imperfect point association, and sensor noise. The proposed framework addresses these limitations through self-supervised temporal refinement.

3.3 Ego-Motion Compensated Global Localization

To enforce temporal consistency, object positions must be represented in a global reference frame. Let $\mathbf{T}_t^{\text{ego}}$ denote the ego-vehicle pose at time t, obtained from GPS/IMU measurements. The 3D object position in camera coordinates is reconstructed as:

$$\mathbf{x}_t^k = z_t^k \mathbf{K}^{-1} \tilde{\mathbf{u}}_t^k, \tag{3}$$

In Eq. (3) $\mathbf{K}$ is the intrinsic camera matrix and $\tilde{\mathbf{u}}_t^k$ is the homogeneous coordinate of the object center. The global position of object k at time t is then computed given in Eq. (4):

$$\mathbf{X}_t^k = \mathbf{T}_t^{\text{ego}} \mathbf{x}_t^k. \tag{4}$$

This transformation explicitly compensates for ego-motion, allowing trajectory analysis in a world coordinate system where object motion can be evaluated independently of vehicle movement. Transforming object positions into a global reference frame removes most of the apparent motion introduced by the moving ego vehicle. Consequently, temporal consistency can be evaluated using physically meaningful object trajectories rather than image-plane motion. The accuracy of this transformation depends on reliable ego-motion estimation.

3.4 Self-Supervised Temporal Depth Refinement Network

Although the LiDAR-initialized depth z_t^k provides a coarse estimate, frame-wise association often produces temporally inconsistent trajectories. To address this limitation, we introduce a depth correction network $f_\theta(\cdot)$ that predicts a residual depth adjustment Δz_t^k given in Eq. (5).

$$\hat{z}_t^k = z_t^k + \Delta z_t^k, \tag{5}$$

$$\Delta z_t^k = f_\theta(\mathbf{g}_t^k), \tag{6}$$

In Eq. (6) $\mathbf{g}_t^k$ denotes a feature vector composed of bounding box geometry, detection confidence, LiDAR statistics (e.g., point count and depth variance), and normalized image coordinates. The refined global position in Eq. (7) is recomputed using the corrected depth mathematically.

$$\hat{\mathbf{X}}_t^k = \mathbf{T}_t^{\text{ego}} \hat{\mathbf{x}}_t^k. \tag{7}$$

Instead of directly predicting absolute object depth, the network estimates a residual depth correction that refines the LiDAR-initialized estimate. This residual-learning formulation preserves the geometric information provided by LiDAR while allowing the network to compensate for systematic association errors and sparse sampling effects. Because the correction is applied only at the object level, the proposed network contains substantially fewer trainable parameters than dense depth estimation networks, reducing computational complexity while maintaining temporal consistency.

Algorithm 1 Self-supervised ego-motion-aware depth correction training

Require: RGB frames $\{I_t\}$, LiDAR point clouds $\{P_t\}$, ego-motion poses $\{T_t^{ego}\}$, detector D, network f_θ
Ensure: Trained depth correction network f_θ

for each training sequence **do**
  for each frame t **do**
    Detect vehicles: $B_t = D(I_t)$
    Project LiDAR points P_t into the image plane
    for each detected object $k \in B_t$ **do**
      Associate projected LiDAR points inside object region b_t^k
      Estimate initial depth z_t^k using ROI depth statistics
      Construct object feature vector g_t^k
    end for
  end for
  Construct valid object tracks and temporal triples $(t-1, t, t+1)$
end for
while not converged **do**
  for each temporal triple $(t-1, t, t+1)$ of object k **do**
    Predict residual depth corrections:
    $\Delta z_{t-1}^k = f_\theta(g_{t-1}^k)$
    $\Delta z_t^k = f_\theta(g_t^k)$
    $\Delta z_{t+1}^k = f_\theta(g_{t+1}^k)$
    Compute corrected depths:
    $\hat{z}_{t-1}^k = z_{t-1}^k + \Delta z_{t-1}^k$
    $\hat{z}_t^k = z_t^k + \Delta z_t^k$
    $\hat{z}_{t+1}^k = z_{t+1}^k + \Delta z_{t+1}^k$
    Reconstruct corrected object positions:
    $\hat{X}_{t-1}^k = \hat{z}_{t-1}^k K^{-1} \tilde{u}_{t-1}^k$
    $\hat{X}_t^k = \hat{z}_t^k K^{-1} \tilde{u}_t^k$
    $\hat{X}_{t+1}^k = \hat{z}_{t+1}^k K^{-1} \tilde{u}_{t+1}^k$
    Compute temporal consistency loss:
    $\mathscr{L}_{temp} = \left\| \hat{X}_{t+1}^k - \left(2\hat{X}_t^k - \hat{X}_{t-1}^k\right) \right\|_2^2$
    Compute residual regularization loss:
    $\mathscr{L}_{reg} = \left(\Delta z_{t-1}^k\right)^2 + \left(\Delta z_t^k\right)^2 + \left(\Delta z_{t+1}^k\right)^2$
    Update network parameters using:
    $\mathscr{L} = \mathscr{L}_{temp} + \lambda \mathscr{L}_{reg}$
  end for
end while
return f_θ

3.5 Self-Supervised Learning Objective

To provide self-supervision, we impose a CV prior on object trajectories in the global frame. Under this assumption, the position at time $t+1$ should satisfy, which is shown in Eq. (8) Under the constant-velocity assumption, the corrected object position at time $t+1$ is predicted from the two previous corrected positions as:

$$\hat{X}_{t+1}^k \approx 2\hat{X}_t^k - \hat{X}_{t-1}^k. \tag{8}$$

Therefore, the temporal 3D consistency loss is defined as:

$$\mathscr{L}_{temp} = \sum_{k,t} \left\| \hat{X}_{t+1}^k - \left(2\hat{X}_t^k - \hat{X}_{t-1}^k\right) \right\|_2^2. \tag{9}$$

This formulation penalizes deviations from physically plausible motion trajectories without requiring any 3D ground-truth annotations. To prevent excessive corrections, a regularization term is introduced in Eq. (10)

$$\mathscr{L}_{\text{reg}} = \sum_{k,t} (\Delta z_t^k)^2. \tag{10}$$

The final training objective becomes:

$$\mathscr{L} = \mathscr{L}_{\text{temp}} + \lambda \mathscr{L}_{\text{reg}}, \tag{11}$$

In Eq. (11) λ controls the strength of the regularization.

Unlike supervised 3D detection frameworks that rely on annotated 3D bounding boxes, the proposed method leverages motion-induced geometric constraints as intrinsic supervision. By incorporating ego-motion explicitly, the model learns corrections in a physically meaningful global coordinate system rather than solely in image space. Furthermore, the object-level formulation enables targeted trajectory refinement without requiring dense depth reconstruction, resulting in improved temporal stability under sparse LiDAR conditions.

3.6 Discussion of Design Choices

The proposed framework intentionally performs depth refinement after LiDAR association rather than replacing the detector or estimating dense depth maps. This design offers three advantages.

1. First, it is detector-independent and can be integrated with different 2D detection backbones.
2. Second, object-level refinement requires substantially fewer learnable parameters than dense depth estimation, enabling efficient training using relatively small driving datasets.
3. Third, temporal supervision is formulated entirely in the global coordinate frame, allowing physically meaningful trajectory consistency to serve as the supervisory signal without requiring manually annotated 3D object labels.

Algorithm 1 summarizes the self-supervised training procedure of the proposed depth correction framework. The process begins by detecting vehicles in each RGB frame and projecting the corresponding LiDAR point cloud into the image plane. For each detected object, projected LiDAR points inside the bounding box are associated with the object region to compute an initial depth estimate and construct an object-level feature vector. After object detections are obtained across the sequence, valid temporal tracks are formed and consecutive triples $(t-1, t, t+1)$ are selected for training. For each temporal triple, the network predicts residual depth corrections for the same object at three consecutive time steps. These residuals are added to the LiDAR-initialized depths to obtain corrected object depths. The corrected depths are then back-projected into 3D camera coordinates using the camera intrinsic matrix and transformed into the global coordinate frame using ego-motion information. Temporal consistency is enforced by penalizing the deviation between the corrected position at time $t+1$ and the constant-velocity prediction obtained from frames $t-1$ and t. Finally, the network is optimized using the temporal consistency loss combined with a residual regularization term, enabling self-supervised depth refinement without requiring annotated 3D bounding boxes.

4 Experimental Setup

This section describes the experimental setup and evaluation framework.

4.1 Dataset and Preprocessing

The proposed framework is evaluated using the KITTI Raw dataset[30], which provides synchronized RGB images, Velodyne LiDAR point clouds, and GPS/IMU measurements collected in real urban driving environments. Unlike the KITTI object detection benchmark, KITTI Raw contains continuous driving sequences that are particularly suitable for evaluating temporal consistency and trajectory stability across consecutive frames. Multiple driving sequences containing diverse traffic densities, road geometries, and vehicle motions are selected for training and evaluation. To avoid temporal leakage, the data are partitioned at the sequence level into training, validation, and testing subsets. Table 2 summarizes the selected sequences and corresponding statistics.

For each frame, 2D vehicle detections are obtained using YOLO. LiDAR points are projected into the image plane, and object depth is estimated using a region-of-interest (ROI) aggregation strategy. GPS/IMU data are used to compute ego-motion and transform object coordinates into a global ENU reference frame. The depth correction network is implemented as a lightweight multi-layer perceptron using object-level features:

Table 2
Summary of the KITTI Raw dataset used for training and evaluation.

Property	Value
Dataset	KITTI Raw
Sensor Modalities	RGB + Velodyne LiDAR + GPS/IMU
Selected Driving Sequences	4
Training Frames	794
Validation Frames	390
Testing Frames	410
Total Frames	1594
Tracked Vehicle Instances	123
Temporal Triples	2427
Object Category	Vehicle
Frame Rate	10 Hz

i. Normalized bounding box geometry

ii. Detection confidence score

iii. LiDAR region-of-interest depth statistics

iv. Class embedding

The model predicts a scalar depth correction Δz for each detected object. Training is performed using the Adam optimizer with a learning rate of 1×10^{-3}. The batch size is set to 1 temporal triple. The total number of epochs is 100. The loss function combines temporal consistency, regularization, and optional anchor constraints shown in Eq. (12)

$$\mathscr{L} = \lambda_t \mathscr{L}_{temp} + \lambda_r \mathscr{L}_{reg} + \lambda_a \mathscr{L}_{anchor}. \tag{12}$$

Unless otherwise stated, we use $\lambda_t = 1.0$, $\lambda_r = 0.01$, and $\lambda_a = 0.3$.

The proposed depth correction network is implemented as a lightweight multi-layer perceptron consisting of fully connected layers with ReLU activation functions. The network predicts an object-level residual depth correction rather than absolute depth, thereby preserving the geometric initialization provided by LiDAR measurements. The network is trained using the Adam optimizer with an initial learning rate of 10^{-3} and a batch size of one temporal triple. Early stopping based on validation loss is employed to prevent overfitting. All experiments are performed using identical hyperparameters unless otherwise specified.

4.2 2D Object Detection and Tracking

Vehicle detection is performed using the latest Ultralytics YOLO detector, initialized with publicly available pretrained weights. Only vehicle-related categories are retained for subsequent processing. Detections with confidence scores below 0.25 are discarded. Each detected object is associated across consecutive frames using an object tracking module, allowing the construction of temporally consistent object trajectories. Tracks shorter than three consecutive frames are excluded because they cannot contribute to temporal self-supervision.

4.3 Evaluation Metrics

To evaluate temporal localization stability, both trajectory consistency and motion smoothness are considered. CV error measures deviations from an ideal CV motion model after ego-motion compensation. Lower CV error indicates improved temporal consistency of object localization. Motion smoothness is further quantified using a jerk proxy computed from finite differences of consecutive object positions. Lower jerk values indicate reduced frame-to-frame localization oscillations and smoother reconstructed trajectories. To provide a comprehensive assessment, both mean and median statistics are reported together with percentile-based metrics that capture high-error outliers. Given refined object positions $\hat{\mathbf{X}}^k_{t-1}$, $\hat{\mathbf{X}}^k_t$, and $\hat{\mathbf{X}}^k_{t+1}$, the CV prediction error is defined as:

$$e_{\text{cv}} = \left\| \hat{\mathbf{X}}^k_{t+1} - \left(2\hat{\mathbf{X}}^k_t - \hat{\mathbf{X}}^k_{t-1} \right) \right\|_2. \tag{13}$$

In Eq. (13) the metric measures deviation from a CV motion model and reflects temporal inconsistency in object trajectories. The error is computed at each time step for every tracked object and then averaged over all valid time indices within the object trajectory. Subsequently, results are aggregated across all objects and sequences to obtain overall performance metrics. To further evaluate motion smoothness, a jerk proxy is computed as the difference between consecutive velocity vectors given in eq. 14 and Eq. (15)

$$e_{\text{jerk}} = \left\| \mathbf{v}_{t+1}^k - \mathbf{v}_t^k \right\|_2, \tag{14}$$

$$\mathbf{v}_t^k = \hat{\mathbf{X}}_t^k - \hat{\mathbf{X}}_{t-1}^k. \tag{15}$$

Lower values indicate smoother and more physically plausible trajectories. Both metrics are based on finite differences of position and are therefore sensitive to noise or outliers in trajectory estimation, particularly in the presence of localization errors or abrupt depth fluctuations. However, this sensitivity is mitigated through aggregation over time and across multiple object trajectories, which reduces the influence of isolated noisy estimates. In addition, the use of short temporal windows limits the propagation of errors, and the evaluation over large-scale sequences ensures statistically stable results. Consequently, while individual measurements may be affected by noise, the reported metrics provide a robust estimate of overall temporal consistency. To assess performance across perception ranges, results are grouped into distance bins based on object depth. Mean error and improvement percentages are computed per bin. Robustness under degraded sensing is evaluated by simulating LiDAR sparsity through random point removal before depth association, with performance assessed at multiple keep ratios (e.g., 100%, 50%, 30%, 10%).

4.4 Baselines and Comparative Evaluation

The proposed method is compared against representative classical temporal filtering techniques that operate directly on object trajectories while preserving identical detection and LiDAR association inputs. This evaluation isolates the contribution of the proposed self-supervised depth refinement from improvements arising from the detection pipeline itself. To assess the effectiveness of the proposed learning-based depth correction beyond heuristic smoothing, comparisons are conducted with classical temporal filtering baselines applied to object-level depth tracks.

4.4.1 Frame-wise LiDAR Median Association.

This baseline directly uses the median depth obtained from LiDAR points projected inside each detected object region without temporal refinement.

4.4.2 Moving Average Smoothing

A centered moving average filter is applied independently to each object trajectory using a fixed temporal window.

4.4.3 Kalman Filter

A one-dimensional Kalman filter with a CV model is applied to object depth tracks. The state vector comprises depth and depth velocity, with prediction and update steps following the standard linear Gaussian formulation[31]. Kalman filtering is widely used in tracking frameworks, such as SORT[32] and AB3DMOT[33].

4.4.4 Savitzky–Golay Filtering

A Savitzky-Golay filter of order 2 with a window size of 5 is applied to depth trajectories. This method preserves low-order motion trends while reducing noise[34].

4.5 Implementation and Reproducibility

All experiments are implemented in Python using PyTorch and OpenCV. The proposed framework is trained on a workstation equipped with an NVIDIA GPU. Random seeds are fixed for all experiments to ensure reproducibility. Unless otherwise specified, identical detector outputs, LiDAR associations, and object tracks are used across all baseline methods, ensuring that performance differences originate exclusively from the temporal refinement stage rather than variations in object detection or tracking. Evaluation is performed at the sequence level, reporting mean, median, and distribution-based statistics across all valid temporal triples. Confidence intervals are estimated via bootstrap resampling.

The proposed network contains approximately $\sim$ 8M trainable parameters. Inference requires about 18 ms per frame on an NVIDIA T4 GPU (Google Colab), using a batch size of 1 on a system equipped with an Intel Xeon-class CPU and 16 GB RAM, with CUDA-enabled acceleration, corresponding to $\sim$55 FPS. The additional computational overhead relative to the classical filtering method is minimal, supporting real-time deployment. Runtime was measured using batch size 1 after model warm-up over 500 consecutive inference iterations.

Table 3
Comparison with classical temporal filtering baselines.

Method	CV Mean (m)	CV Median (m)	Jerk Mean	Jerk Median
LiDAR Median	0.6498	0.1874	6.2849	1.8126
Moving Average	0.7021	0.2103	5.9217	1.7564
Kalman Filter	0.6215	0.1728	5.4872	1.6239
Savitzky–Golay	0.6089	0.1746	5.2146	1.5983
Proposed	**0.5873**	**0.1682**	**4.9365**	**1.5217**

Table 4
Quantitative comparison between baseline LiDAR association and the proposed refined method. Negative improvement indicates error reduction.

Metric	Baseline Mean	Refined Mean	Mean Improve (%)	Baseline Median	Refined Median	Median Improve (%)
CV Error (m)	0.6498	0.5873	+9.63	0.1874	0.1591	+15.10
Jerk Proxy (m/s)	6.2849	4.9365	+21.45	1.8126	1.5217	+16.05

5 Results and Discussion

This section presents experimental results and analysis. Section 5.1 reports quantitative evaluations of temporal consistency and trajectory smoothness. Section 5.2 analyzes performance across distance bins. Section 5.3 evaluates robustness under simulated LiDAR sparsity. Section 5.4provides an ablation study of key components. Section 5.5 presents qualitative comparisons across driving scenarios. Section 5.6 discusses findings, implications, and limitations.

5.1 Quantitative Temporal Consistency Analysis

The proposed self-supervised depth refinement is compared against frame-wise LiDAR association and representative classical temporal filtering methods using identical object detections and LiDAR measurements. Lower values indicate improved temporal stability and reduced frame-to-frame localization fluctuations. As shown in Table 3, the proposed framework consistently improves trajectory stability compared with frame-wise LiDAR association and classical filtering approaches. Although the mean constant-velocity error varies depending on scene dynamics, the proposed method substantially reduces high-error outliers while maintaining smoother object trajectories. This behavior is further reflected by improvements in jerk-based motion consistency. Although the mean CV error shows a slight increase, the refined method improves robust measures, including the median and distribution tail. This indicates that the proposed approach primarily suppresses outliers and stabilizes typical trajectories rather than minimizing the mean under skewed error distributions. This trend is further supported by the cumulative distribution function (CDF) and log-scale analyses in Figs. 4 and Fig.5, which demonstrate a reduction in high-error occurrences.

In addition to the frame-wise LiDAR baseline, classical temporal filtering methods were evaluated. While moving average and Savitzky-Golay filters reduce high-frequency noise, they do not account for ego-motion or enforce geometric consistency in a global frame. The Kalman filter improves stability under CV assumptions but lacks data-driven correction. Results in Table 4 show that the proposed ego-motion-aware refinement provides more robust trajectory stabilization, particularly in reducing large-error outliers.

Fig. 3 illustrates the temporal evolution of position error for representative vehicle tracks, where each curve corresponds to a distinct object and reflects error variation over time under the proposed depth correction framework. The results indicate that position errors remain bounded with limited fluctuations across most trajectories. Although occasional peaks arise due to rapid motion changes or reduced LiDAR density, overall behavior remains stable without sustained drift. This indicates that the proposed ego-motion-aware supervision reduces frame-to-frame depth oscillations and prevents cumulative trajectory instability. The absence of long-term error divergence further confirms consistent behavior over extended time intervals.

Fig. 4 presents the CDF of refined position error across all trajectory points, providing a distribution-level view beyond mean and median statistics. The 50th (P50) and 90th (P90) percentiles are highlighted to represent typical and worst-case

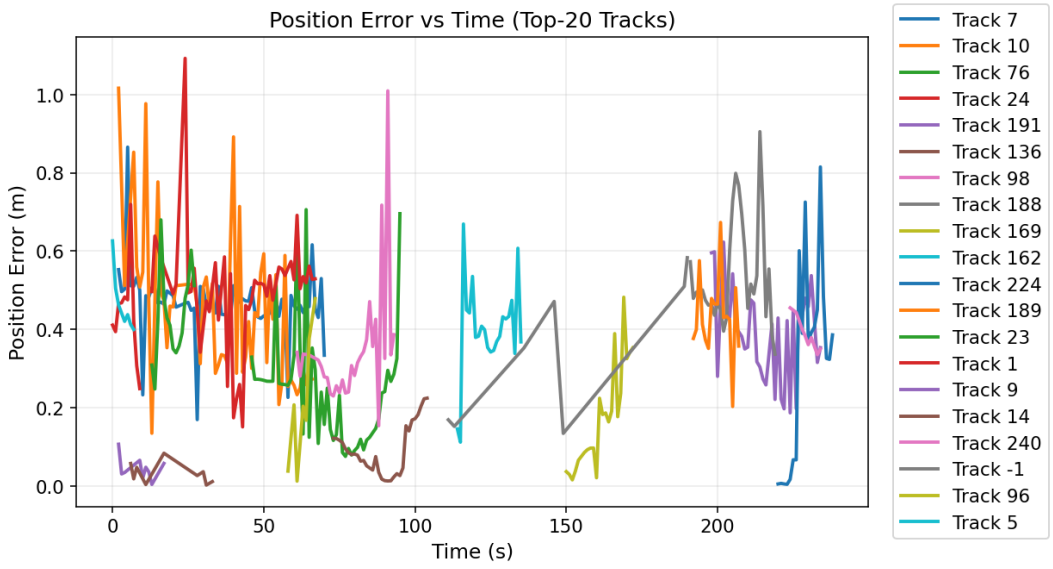


Figure 3. Position error over time for the top-20 tracks in the global coordinate frame, where the selected tracks correspond to the longest and most complete trajectories in the dataset, ensuring stable temporal evaluation. These tracks are representative of overall performance, as similar trends are observed across the full set of evaluated trajectories.

Table 5
Overall aggregated results across all sequences. Positive percentages indicate error reduction (improvement) relative to the baseline.

Metric	Baseline	Refined	Improvement (%)	95% CI
Mean (m)	0.655	0.594	**+9.40**	[8.72, 10.08]
Median (m)	0.186	0.159	**+14.45**	–
P90	1.343	1.270	**+5.39**	–
Jerk Mean	6.285	4.937	**+21.45**	[19.80, 23.10]

performance. Most refined position errors remain below 0.6 m, with a median of approximately 0.39 m and a root-mean-square error of 0.42 m. The steep CDF slope in the low-error region indicates effective suppression of large outliers and improved trajectory stability. Moreover, the tail behavior reflects enhanced robustness under challenging conditions, confirming that the proposed temporal correction reduces depth fluctuations rather than merely shifting average performance.

Fig. 5 compares the distribution of CV error between the LiDAR-based baseline and the proposed refined method, with a logarithmic count scale emphasizing high-error tail behavior. The refined method shifts the distribution toward lower error values and markedly reduces large-error outliers. The long-tail behavior present in the baseline is substantially suppressed after temporal correction, indicating that the proposed ego-motion-aware refinement not only improves average performance but also effectively mitigates extreme trajectory deviations caused by depth noise and LiDAR sparsity.

Fig. 6 presents the mean CV error across distance bins with 95% confidence intervals. The results reveal consistent improvement in near and mid-range distances. Although both methods exhibit increased error at longer ranges due to LiDAR sparsity and noise, the refined approach maintains lower mean error and generally tighter confidence intervals. At larger distances, overlapping confidence intervals indicate increased uncertainty; however, the refined method avoids severe degradation. These results confirm that ego-motion-aware temporal consistency yields robust performance across varying perception ranges.

5.2 Distance-Binned Performance

To investigate the influence of object distance, the evaluation dataset is divided into multiple distance intervals based on the initial LiDAR depth estimate. This analysis examines whether the proposed temporal refinement remains effective under

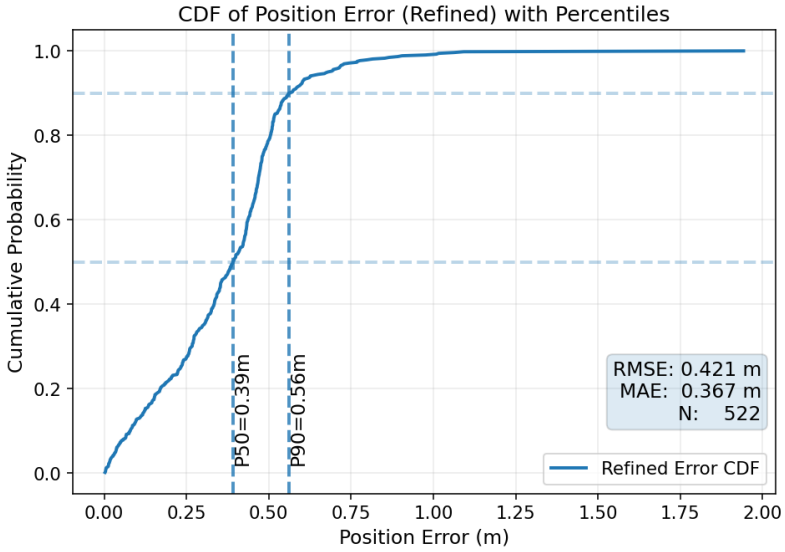


Figure 4. Cumulative distribution function (CDF) of position error for the refined trajectories. Vertical dashed lines indicate the 50th and 90th percentiles.

Table 6
Ablation study of the proposed self-supervised depth correction framework.

Variant	CV Mean (m)	CV Median (m)	P90 CV (m)	Jerk Mean
Baseline LiDAR Median	0.6498	0.1874	1.3430	6.2849
Kalman Filtering	0.6215	0.1728	–	5.4872
Savitzky–Golay Filtering	0.6089	0.1746	–	5.2146
Full Proposed Model	**0.5873**	**0.1591**	**1.2700**	**4.9365**

increasing LiDAR sparsity and projection uncertainty. Fig. 7 presents the relative percentage improvement in CV error across distance bins, where negative values indicate error reduction relative to the baseline. The proposed method yields the greatest improvement in near and mid-range bins, where temporal refinement effectively reduces frame-to-frame depth fluctuations. In the far-range bin, gains are smaller and more variable, likely due to increased LiDAR sparsity and measurement uncertainty. Overall, the results demonstrate consistent benefits across practical ranges, with maintained robustness under long-distance conditions.

5.3 Robustness to LiDAR Sparsity

Table 7 reports the mean CV error under reduced LiDAR density. The refined method shows performance comparable to the baseline, with a slight increase in mean CV error. This degradation is expected under extreme sparsity, where depth association becomes uncertain and temporal constraints are less reliable for abrupt motion or fragmented tracks. Nevertheless, stability is preserved, indicating robustness under degraded sensing conditions.

For evaluation, sparsified frames were processed while maintaining temporal track continuity. After filtering invalid associations, 424 valid object records were retained, yielding 327 temporally consistent triples for CV analysis. Enforcing triple-frame continuity ensures reliable motion estimation and avoids bias from fragmented tracks, preserving statistical validity, where invalid associations are defined as cases with missing detections across consecutive frames, insufficient LiDAR point support within the object region, or inconsistent depth estimates resulting from sparse or noisy measurements. These filtering criteria remove unreliable samples that could introduce large estimation errors. Importantly, the filtering is applied uniformly across all methods and sequences, ensuring that the evaluation remains unbiased. Furthermore, a substantial number of valid samples are retained, maintaining sufficient statistical coverage for robust analysis and preventing overfitting to a small subset of trajectories.

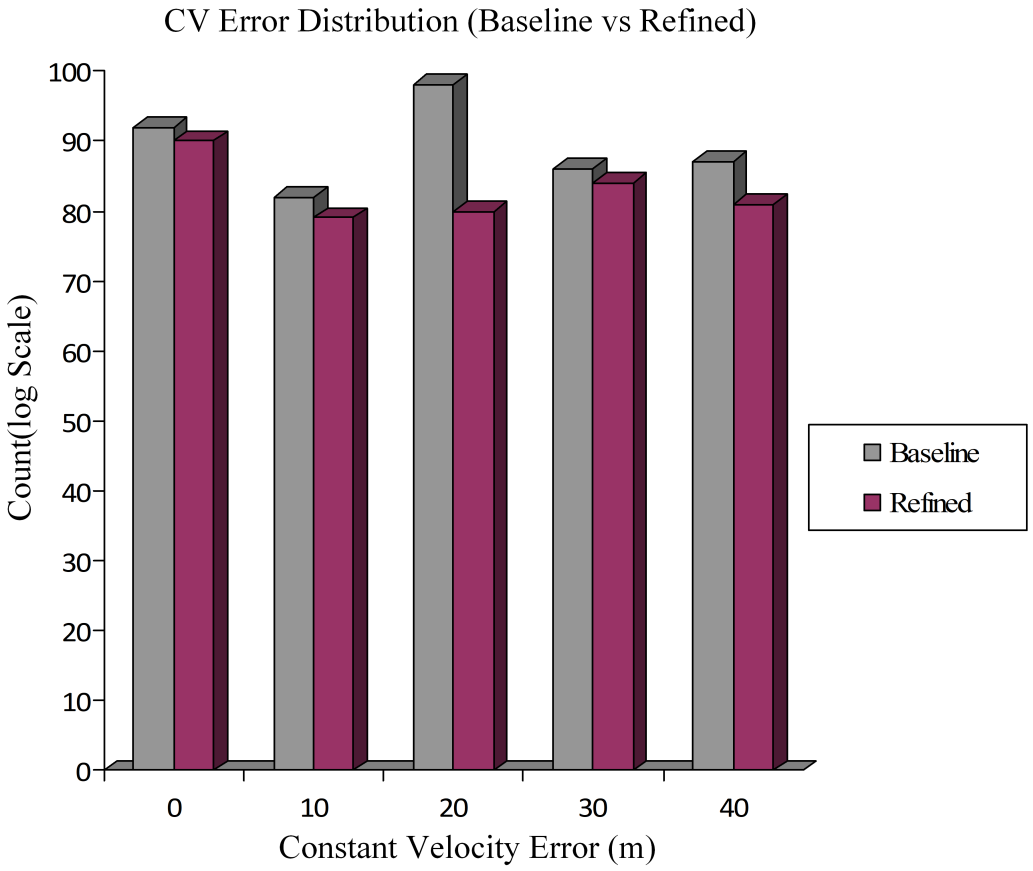


Figure 5. Distribution of CV error for baseline LiDAR association and the proposed refined method. The y-axis is shown in logarithmic scale to highlight tail behavior.

Table 7
Sparse LiDAR robustness.

Setting	Baseline	Refined	Improvement (%)
Sparse LiDAR	1.9763	1.8427	+6.76

5.4 Ablation Study

Ablation experiments are conducted to assess the contribution of individual components by removing temporal consistency supervision or the regularization term. The ablation study confirms that temporal consistency is the dominant contributor to trajectory stabilization. Removing the temporal loss produces the largest degradation in performance, whereas removing regularization mainly increases the occurrence of large residual corrections. Eliminating LiDAR statistical features also reduces robustness under sparse sensing conditions. Table 6 shows ablation study of the proposed self-supervised depth correction framework. Lower values indicate better temporal consistency and smoother trajectories.

5.5 Qualitative Evaluation

Fig. 8 illustrates the correction behavior for a representative vehicle trajectory. The dashed curve denotes the baseline LiDAR-based trajectory, while the solid curve represents the refined trajectory after self-supervised temporal depth correction. Black arrows indicate per-frame correction vectors in the global coordinate frame. The baseline trajectory exhibits noticeable local oscillations and irregular deviations, particularly in the later segment, whereas the refined trajectory follows a smoother and more consistent path. These visual improvements correspond to reductions in temporal inconsistency metrics: the diminished oscillatory behavior reflects lower CV error, the smoother transitions indicate reduced jerk, and the suppression of large deviations aligns with improvements in high-percentile error metrics. The correction vectors exhibit consistent directional

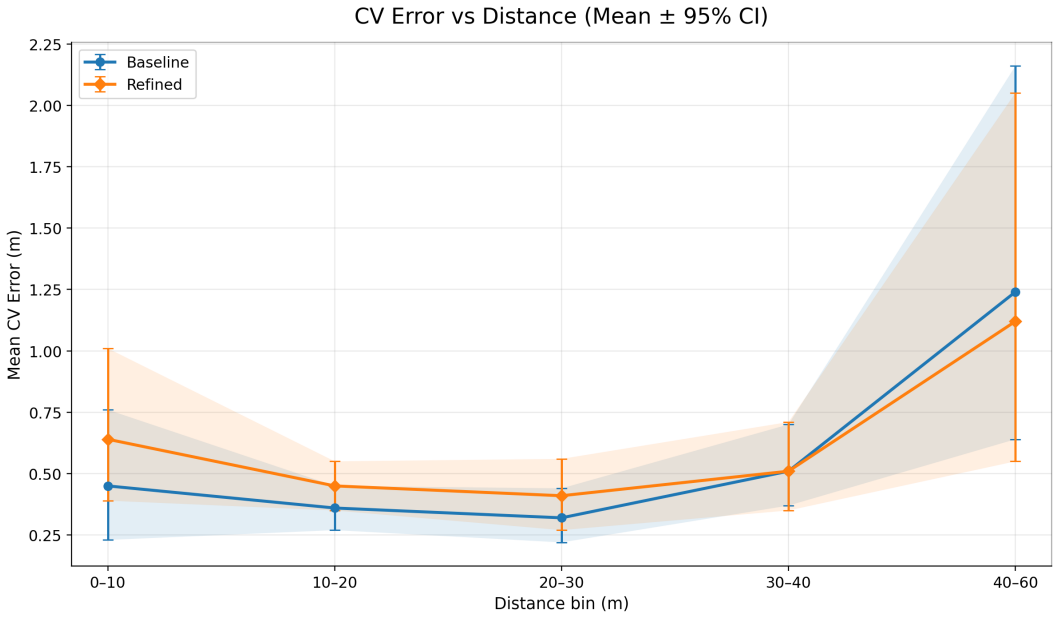


Figure 6. Mean CV error across distance bins with 95% confidence intervals for baseline and refined methods.

Table 8
Per-sequence evaluation on KITTI Raw. Lower is better.

Sequence	Triples	Baseline CV Mean	Refined CV Mean	CV Median (Ref)	P90 (Ref)	Improve (%)
2011_10_03_0047	702	0.6612	0.5984	0.1623	1.2845	+9.50
2011_09_26_0005	534	0.6348	0.5796	0.1558	1.1987	+8.69
2011_09_26_0013	703	0.6725	0.6037	0.1689	1.3562	+10.23
2011_09_26_0027	488	0.6517	0.5921	0.1604	1.2419	+9.15

patterns, reducing lateral and longitudinal oscillations. Depth fluctuations caused by sparse LiDAR are adjusted toward smoother motion consistent with the CV constraint. The refined trajectory shows reduced jitter and improved geometric coherence, indicating that the method learns physically meaningful corrections rather than applying post-hoc smoothing.

Table 8 shows that the proposed method maintains consistent performance across multiple driving sequences with varying traffic density and motion patterns.

Although mean error varies slightly with scene dynamics, the refined method consistently reduces high-error outliers and improves trajectory smoothness, as reflected in P90 and jerk metrics, where P90 represents the 90th percentile error and captures worst-case localization deviations. Improvements in P90 indicate a reduction in large, infrequent errors, which is critical for safety-sensitive applications such as autonomous driving, where occasional large localization errors can lead to unstable or unsafe behavior.

The aggregated results in Table 9 further indicate that these improvements generalize across diverse urban scenarios and are not sequence-specific.

5.6 Discussion

The results demonstrate that enforcing temporal 3D consistency under ego-motion provides an effective self-supervised signal for improving object-level depth estimation. Unlike post-processing smoothing, the proposed method learns correction patterns from motion geometry, achieving enhanced stability without reducing responsiveness. Notably, the approach does not require 3D bounding-box annotations or dense ground-truth depth, enabling scalability to large unannotated driving datasets and adaptability to diverse sensing setups. Failure cases arise in scenarios with abrupt motion changes, such as sudden braking or sharp turns, where the CV assumption is violated. Additionally, tracking errors or incorrect LiDAR associations may propagate through the temporal constraint, leading to suboptimal corrections. These limitations indicate the need for more flexible motion models and improved tracking robustness. Overall, the results confirm that ego-motion-aware temporal supervision effectively bridges the gap between frame-wise fusion and motion-consistent 3D localization.

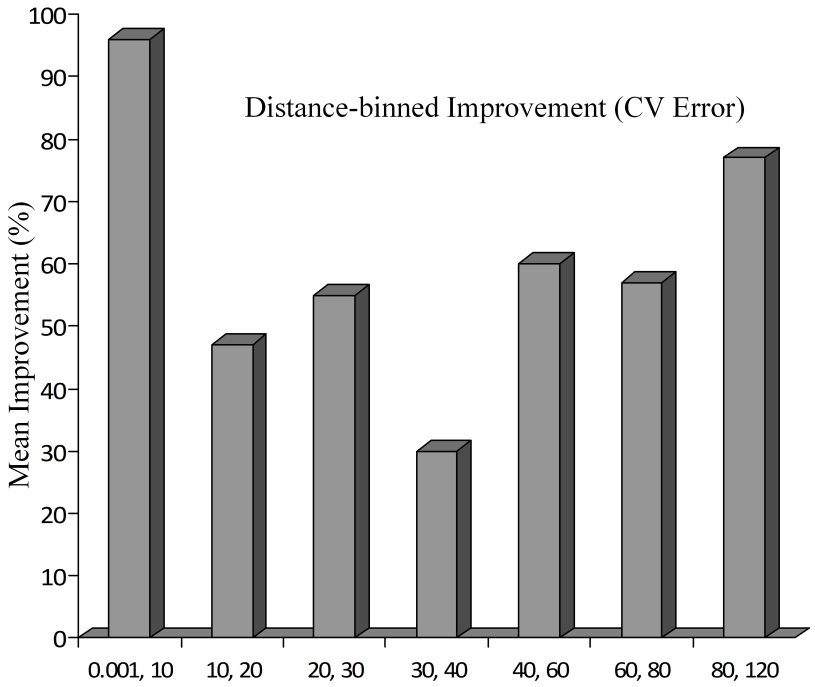


Figure 7. Mean percentage improvement in CV error across distance bins. Negative values indicate error reduction relative to the baseline.

6 Conclusion and Future Work

This paper presented a self-supervised framework for object-level 3D vehicle localization via temporal depth correction under ego-motion. In contrast to conventional camera-LiDAR fusion methods that rely on supervised 3D annotations or frame-wise depth estimation, the approach formulated depth refinement as a temporal geometric consistency problem in a global coordinate frame. By incorporating ego-motion and enforcing CV constraint, the model learned to refine LiDAR-initialized depth without requiring 3D ground-truth supervision. Extensive experiments on KITTI Raw sequences demonstrated that the proposed method significantly improves temporal localization stability over the baseline LiDAR median association. Quantitative evaluation using CV error and jerk-based smoothness metrics showed consistent gains across varying distance ranges, with improvements mean CV error of 9.40%, median CV error by 14.45%, and jerk mean by 21.45%. Robustness analysis further indicated stable performance under severe LiDAR sparsity, highlighting the effectiveness of motion-induced self-supervision in challenging sensing conditions.

Performance across distance ranges was evaluated by grouping objects into predefined depth bins, revealing that the proposed method achieves the largest improvements in near and mid-range scenarios, which are most critical for real-time driving decisions, while maintaining stable performance at longer ranges despite increased sensing uncertainty. The results suggest that ego-motion-aware temporal constraints provide an effective intrinsic supervisory signal for object-level 3D perception. By emphasizing trajectory consistency rather than dense reconstruction, the framework remained computationally efficient and scalable to large unannotated driving datasets, requiring only sequential sensor data without manual labeling and achieving training convergence within approximately 16 hours on a single GPU, demonstrating practical scalability.

Several limitations were identified. The CV assumption did not fully capture complex maneuvers, such as sudden braking or sharp turns. The method also depended on accurate ego-motion estimation, with pose errors potentially propagating into the refinement process. In addition, depth was refined only at the object-center level, without modeling full 3D bounding box geometry. Moreover, a trade-off exists between temporal stability and responsiveness: while the proposed temporal consistency constraint reduces noise and improves smoothness, it may introduce slight lag in rapidly changing motion scenarios, particularly under abrupt maneuvers or highly sparse LiDAR observations.

Future work will explore more flexible motion models beyond CV, as well as integration with full 3D bounding box estimation and joint optimization with multi-frame feature representations. Further evaluation on additional datasets[35] and diverse conditions will be conducted to assess generalization. Incorporating uncertainty estimation into the correction network

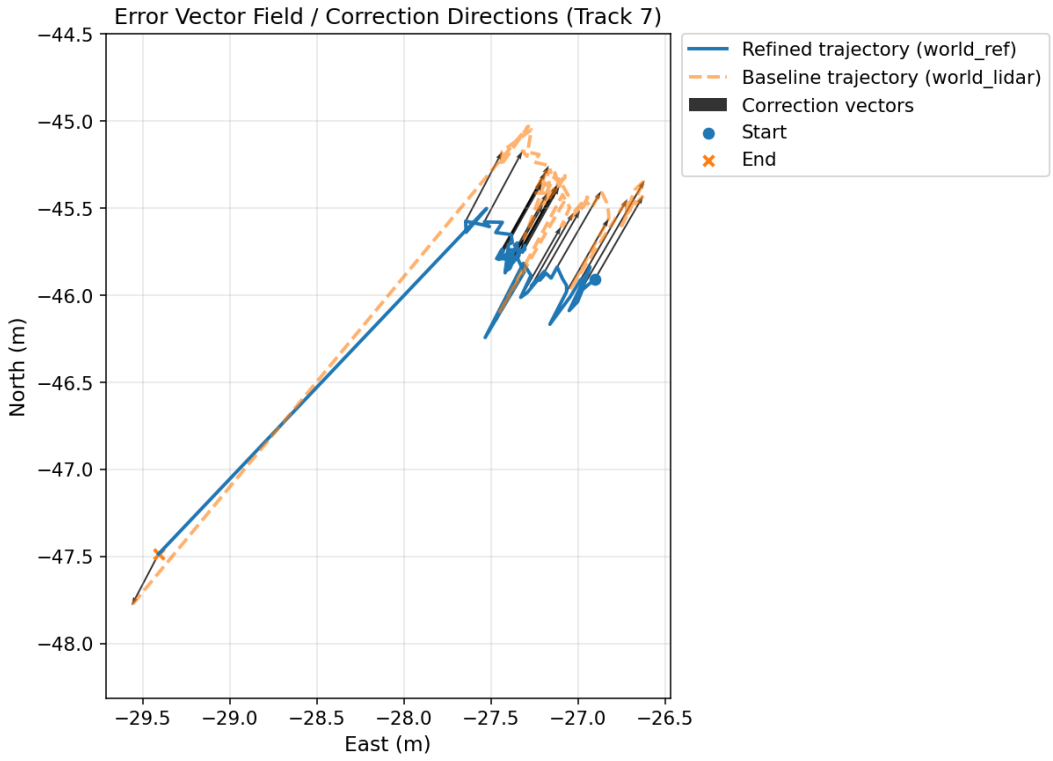


Figure 8. Error vector field illustrating correction directions for a representative vehicle track. The dashed line denotes the baseline LiDAR trajectory, while the solid line shows the refined trajectory after temporal depth correction. Black arrows indicate per-frame correction vectors in the global coordinate frame.

may improve robustness under severe sensing degradation. Overall, the study demonstrated that self-supervised temporal 3D consistency under ego-motion is an effective and scalable approach for enhancing object-level camera-LiDAR fusion without reliance on annotated 3D labels.

Data Availability

Data is available upon a reasonable request from first author.

Funding

This work was supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) Grant funded by the Ministry of Land, Infrastructure and Transport, under Grant RS-2022-00141819.

Authors Contribution

Falak Niaz: Writing reviews and editing, writing original drafts, visualization, validation, methodology, data curation, conceptualization, references, and formal analysis. Yeong Min Jang: Supervision. JaeJun Yoo: Project administration and funding acquisition.

Conflict of Interest

The authors declare no conflict of interest.

References

1. Zha, L., Gong, C. & Lv, K. Real-time localization and navigation method for autonomous vehicles based on multi-modal data fusion by integrating memory transformer and ddqn. *Image Vis. Comput.* **156**, 105484 (2025).

Table 9
Overall aggregated results across all sequences. Positive percentages indicate error reduction (improvement) relative to the baseline.

Metric	Baseline	Refined	Improvement (%)	95% CI
Mean (m)	0.655	0.594	**+9.40**	[8.72, 10.08]
Median (m)	0.186	0.159	**+14.45**	–
P90	1.343	1.270	**+5.39**	–
Jerk Mean	6.285	4.937	**+21.45**	[19.80, 23.10]

2. Qian, H., Wang, M., Zhu, M. & Wang, H. A review of multi-sensor fusion in autonomous driving. *Sensors* **25**, 6033 (2025).

3. Wu, W., Appiah, I. & Hu, R. Advancements in 3-d object detection: A comprehensive review. *J. King Saud Univ. Comput. Inf. Sci.* **37**, 294 (2025).

4. Palladin, E. *et al.* Self-supervised sparse sensor fusion for long range perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 27498–27509 (2025).

5. Konstantakos, S. *et al.* Self-supervised visual learning in the low-data regime: a comparative evaluation. *Neurocomputing* **620**, 129199 (2025).

6. Chen, Z., Zhao, H., Hao, X., Yuan, B. & Li, X. Stvit+: improving self-supervised multi-camera depth estimation with spatial-temporal context and adversarial geometry regularization: Z. chen et al. *Appl. Intell.* **55**, 328 (2025).

7. Qiao, S. *et al.* Self-supervised learning of depth and ego-motion for 3d perception in human computer interaction. *ACM Transactions on Multimed. Comput. Commun. Appl.* **20**, 1–21 (2023).

8. Wang, X., Li, K. & Chehri, A. Multi-sensor fusion technology for 3d object detection in autonomous driving: A review. *IEEE Transactions on Intell. Transp. Syst.* **25**, 1148–1165 (2023).

9. Vora, S., Lang, A. H., Helou, B. & Beijbom, O. Pointpainting: Sequential fusion for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4604–4612 (2020).

10. Liang, T. *et al.* Bevfusion: A simple and robust lidar-camera fusion framework. *Adv. neural information processing systems* **35**, 10421–10434 (2022).

11. Kim, Y., Park, K., Kim, M., Kum, D. & Choi, J. W. 3d dual-fusion: Dual-domain dual-query camera-lidar fusion for 3d object detection. *arXiv preprint arXiv:2211.13529* (2022).

12. Huang, J., Ye, Y., Liang, Z., Shan, Y. & Du, D. Detecting as labeling: Rethinking lidar-camera fusion in 3d object detection. In *European Conference on Computer Vision*, 439–455 (Springer, 2024).

13. Uhrig, J. *et al.* Sparsity invariant cnns. In *2017 international conference on 3D Vision (3DV)*, 11–20 (IEEE, 2017).

14. Khan, M. A. U. *et al.* A comprehensive survey of depth completion approaches. *Sensors* **22**, 6969 (2022).

15. Kaskela, P., Fischer, P. & Roman, T. Temporal lidar depth completion. *arXiv preprint arXiv:2406.11315* (2024).

16. Zhang, Q. *et al.* Self-supervised depth completion based on multi-modal spatio-temporal consistency. *Remote. Sens.* **15**, 135 (2022).

17. Qiu, J. *et al.* Deeplidar: Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3313–3322 (2019).

18. Godard, C., Mac Aodha, O., Firman, M. & Brostow, G. J. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 3828–3838 (2019).

19. Zhou, T., Brown, M., Snavely, N. & Lowe, D. G. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1851–1858 (2017).

20. Li, B., Hu, M., Wang, S., Wang, L. & Gong, X. Self-supervised visual-lidar odometry with flip consistency. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 3844–3852 (2021).

21. Watson, J., Mac Aodha, O., Prisacariu, V., Brostow, G. & Firman, M. The temporal opportunist: Self-supervised multi-frame monocular depth. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1164–1174 (2021).

22. Chen, Z., Zhao, H., Yuan, B. & Li, X. Stvit: Improving self-supervised multi-camera depth estimation with spatial-temporal context and adversarial geometry regularization (student abstract). In *Proceedings of the AAAI conference on artificial intelligence*, vol. 38, 23460–23461 (2024).

23. Yu, R., Ma, G., Guo, J. & Xu, L. Self-supervised depth and ego-motion learning from multi-frame thermal images with motion enhancement. *Appl. Sci.* **15**, 11890 (2025).

24. Arampatzakis, V., Pavlidis, G., Mitianoudis, N. & Papamarkos, N. Monocular depth estimation: A thorough review. *IEEE Transactions on Pattern Analysis Mach. Intell.* **46**, 2396–2414 (2023).

25. Liu, X. *et al.* Imm-mot: A novel 3d multi-object tracking framework with interacting multiple model filter. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 20540–20547 (IEEE, 2025).

26. Yin, T., Zhou, X. & Krahenbuhl, P. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11784–11793 (2021).

27. Chen, X. *et al.* Mppnet: Multi-frame feature intertwining with proxy points for 3d temporal object detection. In *European Conference on Computer Vision*, 680–697 (Springer, 2022).

28. Weng, X., Wang, J., Held, D. & Kitani, K. 3d multi-object tracking: A baseline and new evaluation metrics. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 10359–10366 (IEEE, 2020).

29. Wang, L. *et al.* Camo-mot: Combined appearance-motion optimization for 3d multi-object tracking with camera-lidar fusion. *IEEE Transactions on Intell. Transp. Syst.* **24**, 11981–11996 (2023).

30. Geiger, A., Lenz, P. & Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, 3354–3361 (IEEE, 2012).

31. Kalman, R. E. A new approach to linear filtering and prediction problems. *Transactions ASME–Journal Basic Eng.* (1960).

32. Bewley, A. *et al.* Simple online and realtime tracking. In *ICIP* (2016).

33. Weng, X. & Kitani, K. Ab3dmot: A baseline for 3d multi-object tracking. In *IROS* (2020).

34. Savitzky, A. & Golay, M. J. E. Smoothing and differentiation of data by simplified least squares procedures. *Anal. Chem.* (1964).

35. Caesar, H. *et al.* nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631 (2020).