Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.076652

REVIEW

# 3D Single Object Tracking in Point Clouds: A Review

**Yihao Kuang**[1,2], **Hong Zhang**[1,2], **Jiaqi Wang**[1,2], **Lingyu Jin**[1,2] **and Bo Huang**[1,2,*]

[1]College of Optoelectronic Engineering, Chongqing University, Chongqing, China
[2]Key Laboratory of Optoelectronic Technology and Systems of the Education Ministry of China, Chongqing University, Chongqing, China

*Corresponding Author: Bo Huang. Email: huangbo0326@cqu.edu.cn

Received: 24 November 2025; Accepted: 30 January 2026; Published: 09 April 2026

**ABSTRACT:** 3D single object tracking (SOT) based on point clouds is a fundamental task for environmental perception in autonomous driving and dynamic scene understanding in robotics. Recent technological advancements in this field have significantly bolstered the environmental interaction capabilities of intelligent systems. This field faces persistent challenges, including feature degradation induced by point cloud sparsity, representation drift caused by non-rigid deformation, and occlusion in complex scenarios. Traditional appearance matching methods, particularly those relying on Siamese networks, are severely constrained by point cloud characteristics, often failing under rapid motions or structural ambiguities among similar objects. In response, the research paradigm has progressively evolved toward motion-centric modeling approaches. These emerging frameworks utilize spatio-temporal joint modeling and geometric shape completion to attain notable performance gains. Furthermore, the incorporation of attention mechanisms and State Space Model (SSM) has enabled more effective multi-scale spatio-temporal feature association, which is particularly beneficial for long-term tracking scenarios. To the best of our knowledge, this is the first comprehensive survey dedicated to 3D single object tracking in point clouds. We provide a detailed analysis of current tracking methods, scrutinizing their limitations regarding multi-object interference and analyzing the trade-off between accuracy and computational efficiency. Finally, we discuss potential future directions, including the development of lightweight models for edge deployment and the integration of cross-modal fusion strategies.

**KEYWORDS:** Point cloud; 3D single object tracking; autonomous driving; sparsity; spatio-temporal feature

## 1 Introduction

### *1.1 Motivation and Background*

3D SOT, a core task in computer vision and autonomous driving, aims to continuously localize target objects across consecutive point cloud frames using their initial states. The proliferation of 3D sensing technologies (e.g., LiDAR) has established point clouds as a critical data source for object tracking, primarily due to their illumination invariance and inherent geometric information. However, inherent challenges including sparsity, spatial disorder, and non-uniform density distribution fundamentally complicate point cloud processing. Furthermore, object deformation during motion, dynamic occlusion, and interference from inter-class similarity present ongoing tracking challenges.

In practical scenarios, sparsity manifests acutely; for instance, long-distance targets (e.g., pedestrians 100 m away) may contain only 20–50 points, making it difficult to extract stable features; in rainy or foggy weather, LiDAR point cloud density can drop by 60%, blurring target contours. Spatial disorder, referring

to the fact that point clouds have no fixed order, renders traditional 2D feature extraction methods—dependent on grid structures—inapplicable, necessitating specialized architectures compatible with disorder. Non-uniform density distribution further causes near-dense and far-sparse biases in feature extraction, affecting global localization accuracy. Furthermore, object deformation during motion, dynamic occlusion, and interference from inter-class similarity present ongoing tracking challenges—deformation leads to mismatches between templates and current frames; occlusion may cause target point clouds to disappear entirely; and inter-class similarity easily triggers mistracking in complex traffic flows.

Early approaches predominantly adapted 2D visual similarity matching paradigms, determining object positions through template-search region correspondences [1,2]. However, 2D methods rely heavily on texture information, which is scarce in point clouds, leading to inherent mismatches with 3D point cloud characteristics. For example, 2D cross-correlation operations fail to handle point cloud disorder, resulting in unstable matching in sparse scenarios. Although conceptually straightforward, these methods were fundamentally limited by point cloud sparsity, exhibiting significant difficulty in discriminating geometrically similar distractors. Moreover, the reliance on handcrafted or computationally intensive bounding box estimation techniques—such as Kalman filtering [3]—introduces significant computational overhead. Traditional Kalman filtering-based methods often require $\mathcal{O}(n^3)$ matrix operations per frame, which poses a major bottleneck for real-time deployment in resource-constrained environments.

With the rapid growth of point cloud data and advancements in computational resources, end-to-end learning frameworks have made significant strides by introducing collaborative voting schemes for object center candidate generation [4]. For instance, VoteNet [5] aggregates sparse points into object centers through point-wise voting, effectively addressing the sparse-to-dense localization bottleneck. This approach notably improves both localization accuracy and computational efficiency, thereby reinforcing the prevalence of matching paradigms. Subsequent refinements have further enhanced feature matching via employing voxel-based representations [6], and augmenting local geometric features [7]. Despite these advances, fundamental challenges remain—particularly in mitigating tracking drift caused by inherent point cloud sparsity. Even state-of-the-art methods exhibit a 30% higher drift rate in extremely sparse scenarios (≤30 points) compared to dense scenarios.

To overcome the constraints of matching paradigms, motion-centric modeling has emerged as a promising alternative. These techniques explicitly model inter-frame object motion dynamics, shifting focus from appearance similarity to motion patterns—often through techniques such as pose transformation estimation via foreground point segmentation [8]. This methodology exhibits superior robustness under occlusion and point cloud sparsity conditions. However, dependence on local motion cues from neighboring frames fundamentally limits long-term motion pattern capture. Moreover, segmentation inaccuracies propagate pose estimation errors, leading to cumulative drift. Recent breakthroughs have begun to mitigate these issues by integrating multimodal data fusion [9], temporal context modeling [10], and Transformer architectures [11], which significantly enhance feature extraction and correspondence accuracy. Simultaneously, progress in point cloud representation learning and lightweight network design enables novel accuracy-efficiency tradeoffs [12].

To establish a rigorous research boundary, we adopt the formal problem definition proposed by Hodaň et al. [13]. Mathematically, the 3D SOT task aims to estimate the optimal bounding box $B_t$ of a target in the current frame $P_t$, given the initial state $B_0$ and a sequence of historical observations $\mathcal{P}_{0:t-1}$, by maximizing the posterior probability $P(B_t|B_{t-1}, P_t)$. Within this theoretical framework, this review explicitly restricts its scope to 'complex dynamic scenarios' in autonomous driving, which are characterized by high sparsity (objects with fewer than 50 points), severe occlusion, and non-rigid deformation. Consequently, we focus

exclusively on tracking algorithms designed for unstructured and sparse LiDAR point clouds, extending the discussion from foundational Siamese-based frameworks to emerging Motion-Centric and SSM approaches.

Despite the rapid growth of this field, a systematic survey dedicated to 3D SOT is notably absent. Existing reviews primarily focus on 3D object detection or 2D tracking, often overlooking the specific challenges of temporal coherence in sparse point clouds. Previous surveys on multi-object tracking also differ significantly in problem formulation. Therefore, a comprehensive analysis of 3D SOT algorithms is urgently needed.

### *1.2 Literature Search and Selection Criteria*

To ensure a rigorous and objective consolidation, we implemented a systematic literature search and filtering methodology. The retrieval was conducted across major academic databases, including IEEE Xplore, Web of Science, Google Scholar, and the Computer Vision Foundation Open Access, covering the period from January 2019 to January 2026. The search strategy utilized Boolean logic with keyword combinations such as "Point Cloud," "LiDAR," "3D," "Single Object Tracking," and "Deep Learning." This initial phase yielded over 300 candidate entries.

To isolate the most significant contributions, we applied a strict multi-stage screening process. First, the scope was restricted exclusively to 3D SOT, excluding studies focused solely on 2D tracking or Multi-Object Tracking unless they offered specific contributions to the SOT domain. Second, priority was given to peer-reviewed articles published in high-impact journals (e.g., IEEE TPAMI, TITS) and top-tier conferences (e.g., CVPR, ICCV, ECCV, AAAI), with a requirement that the methods be validated on standard benchmarks. Finally, to capture the rapid evolution of emerging architectures like Transformers and SSMs, we emphasized timeliness, with approximately 85% of the selected studies published between 2021 and 2026. Based on these criteria, 77 key publications were ultimately selected for in-depth analysis in this review.

### *1.3 Organization of the Paper*

This survey provides a comprehensive and systematic analysis of the technological evolution in point cloud object tracking. Over the past few years, the field has transitioned from early appearance-based models to more advanced architectures that incorporate motion reasoning and spatio-temporal context modeling. To ground the discussion, a representative pipeline commonly adopted in modern 3D tracking systems is illustrated in Fig. 1. It highlights the classical Siamese network structure, which forms the basis for many current methods, along with its typical extensions in feature extraction, similarity matching, and template updating.

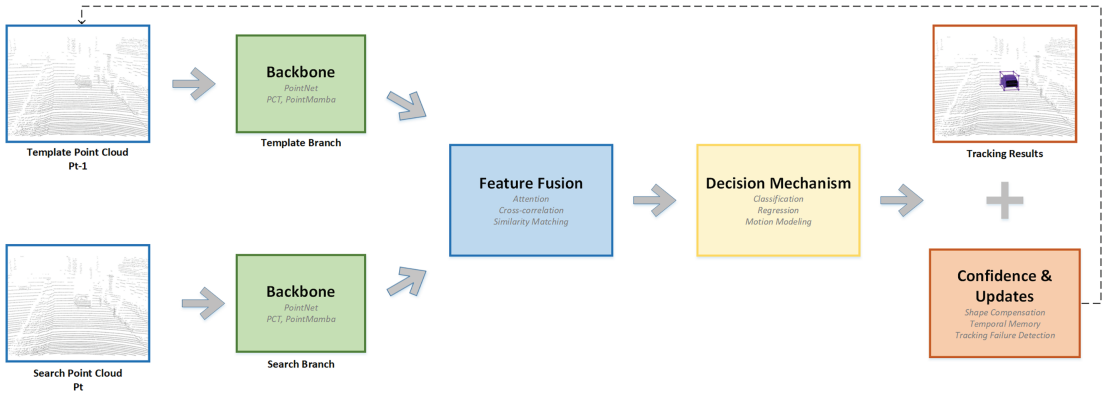


**Figure 1:** General siamese network framework for 3D point cloud object tracking.

Corresponding to the general Siamese-based tracking framework, the structure of this paper is organized as follows: (1) Backbone Networks: This section reviews the development of feature extraction architectures, from early hand-engineered frameworks and voxel-based methods to modern Transformer and Mamba-based models, analyzing their capabilities in capturing geometric structures and spatio-temporal relationships. (2) Decision Mechanisms: We categorize and compare three core paradigms—similarity matching, motion modeling, and attention mechanisms—elucidating their respective strengths in handling appearance variations, motion dynamics, and long-range dependencies. (3) Model Update Strategies: This part focuses on dynamic template optimization, including shape compensation, temporal memory management, and adaptive parameter adjustment, exploring how these strategies mitigate tracking drift under occlusion and sparsity. (4) Experiments: A detailed analysis of mainstream metrics and datasets (KITTI [14], NuScenes [15]) is provided to establish a standardized performance comparison framework. (5) Discussion: This section synthesizes the technological evolution of tracking paradigms and analyzes persistent bottlenecks. (6) Conclusion: This section summarizes the systematic review of 3D single object tracking and outlines future directions.

To the best of our knowledge, this is the first comprehensive survey dedicated to point cloud object tracking. Its key contributions are fourfold: (1) Systematic Taxonomy: We implement a unified classification for tracking paradigms, clarifying the evolutionary logic from 2D adaptation to 3D-specific innovations. As this is a review paper, we do not propose any novel framework. (2) In-Depth Technical Analysis: By dissecting core modules (feature extraction, decision-making, updates), we reveal the theoretical limitations of each method and their performance bottlenecks under real-world constraints. (3) Benchmark Synthesis: A cross-dataset comparison of the state-of-the-art methods is conducted to quantify the effectiveness of different strategies under varying scenarios. (4) Research Outlook: By synthesizing theoretical advances and empirical results, this survey aims to serve as a foundational reference for researchers and engineers in the field, facilitating the development of next-generation point cloud tracking technologies. It is important to clarify that this paper serves as a comprehensive taxonomy and performance analysis of existing methods. While we systematically synthesize the evolution of the field, this work does not propose a novel tracking algorithm but rather consolidates recent years of developments to guide future research expectations.

## 2 Backbone

As a fundamental representation of 3D geometric data, point cloud processing demands efficient backbone network development, as the ability to extract discriminative features directly determines the performance of downstream tasks such as object tracking, segmentation, and reconstruction. Unlike structured 2D images, point clouds are inherently unordered, sparse, and irregularly distributed, posing unique challenges for feature learning—including maintaining permutation invariance, capturing local geometric patterns, and balancing computational efficiency with representation capacity [16]. The evolution of 3D point cloud backbones has thus been driven by addressing these challenges, with distinct stages reflecting advancements in modeling paradigms, as visualized in Fig. 2.

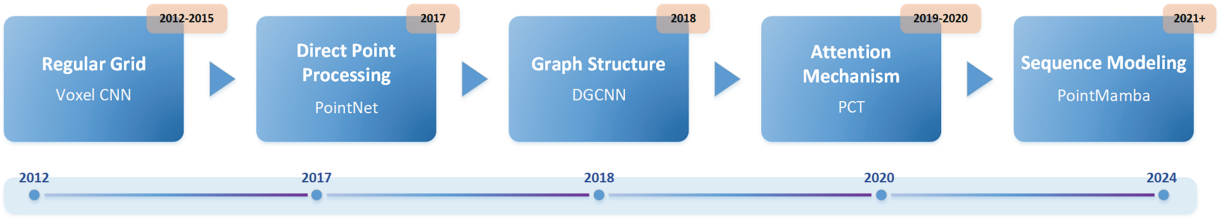


**Figure 2:** The evolution process of the backbone.

### 2.1 Regular Grid

Early approaches addressed the unstructured nature of point clouds by leveraging voxelization, a process that maps 3D points to regular grid spaces [17]. This enabled the adaptation of 2D CNN architectures for spatio-temporal motion modeling, offering a pragmatic solution to data disorder [18,19]. Nevertheless, voxelization inherently introduced quantization artifacts, degrading geometric fidelity and compromising the representation of fine-grained motion trajectories—limitations that motivated subsequent innovations.

### 2.2 Direct Point Processing

The transition to direct point processing represented a paradigm shift in computational efficiency. PointNet [20] pioneered this by using shared multi-layer perceptron (MLP) for permutation invariance, though it failed to capture local structures. PointNet++ [21] addressed this by introducing a hierarchical framework with sampling and grouping, establishing a standard for local feature extraction. However, its recursive sampling creates computational bottlenecks.

Subsequent works focused on refining local feature aggregation. PointCNN [22] introduced X-transformation to enable convolution on unordered points. To enhance geometric reasoning, Relation-Shape CNN [23] and PointWeb [24] modeled inter-point relations and contextual interactions. FoldingNet [25] extended this line of work by introducing folding operations to generate point clouds from latent spaces, achieving state-of-the-art (SOTA) performance in shape completion tasks. Meanwhile, PointConv [26] and PAConv [27] improved computational efficiency by learning dynamic weight functions for irregular point densities.

In summary, point-based backbones process raw coordinates directly, preserving fine-grained geometric details essential for precise localization. However, their reliance on iterative sampling and grouping often creates a computational bottleneck.

### 2.3 Graph Structure

Graph neural networks naturally align with point cloud topology for modeling point cloud topology, given their capacity to capture pairwise relationships. DGCNN [28] pioneered this direction with dynamic graph convolution, constructing k-NN graphs in feature space and learning edge representations via EdgeConv [29]. While this significantly enhanced local geometric modeling, dynamic graph updates resulted in quadratic computational complexity as point counts increased. Subsequent works mitigated this issue: SpiderCNN [30] designed adaptive neighborhood expansion strategies, and Point-GNN [31] refined graph-based message passing to better capture structural dependencies.

### 2.4 Attention Mechanism

Attention-based architectures revolutionized point cloud processing by enabling adaptive feature weighting. Point-BERT [32] introduced a novel pretraining paradigm through Masked Point Modeling.

Transformers [33], with their capacity to model global context, were adapted to point clouds by PCT [34]—the first work to integrate standard Transformer architectures. To handle spatial unorderedness, PCT [34] introduced coordinate embedding (injecting 3D positional information via encoding) and offset attention (dynamically adjusting geometric offsets in attention calculations), thereby strengthening the perception of spatial relationships. Swin3D [35] optimized Transformer efficiency for 3D scenes by adopting window-based attention (inspired by 2D vision): it partitions point clouds into spatial windows, computes

self-attention within windows, and uses window shifting for cross-window interaction, balancing local geometric accuracy and computational cost. Point Transformer [36] further reduced complexity by restricting attention to overlapping windows, thereby striking a finer balance between performance and efficiency.

### *2.5 Sequence Modeling*

Sequence-based methods transformed unordered point clouds into ordered structures, thereby facilitating spatio-temporal modeling. Point2Sequence [37] pioneered this paradigm by converting points into sequences to enhance the modeling of dynamic scene continuity.

Recent advances have leveraged state-space models: PointMamba [38] introduced Mamba architectures to point cloud processing by using Z-order curve partitioning to map unordered points to spatially localized sequences. Combined with bidirectional selective scanning, it maintained spatio-temporal modeling capabilities while reducing GPU memory consumption by over 60%. Its MAE-style pretraining further enhanced the representation of geometric structures. SMamba [39] optimized this framework for sparse point clouds by employing adaptive sparsity-aware mechanisms to boost the efficiency of long-range feature modeling. Additionally, submanifold sparse convolutions mitigated the "near-dense, far-sparse" bias, which is critical for long-range tracking tasks.

## 3 Decision Mechanisms

Driven by breakthroughs in deep learning, the decision paradigms of 3D SOT have gradually evolved from traditional similarity matching to temporal attention, thus giving rise to diverse technical pathways. Fig. 3 illustrates the classification of decision mechanisms. This section focuses on the decision mechanisms of tracking algorithms, systematically analyzing mainstream methods from the perspectives of similarity matching, motion modeling, and Transformer architectures. Through experimental data, it reveals the advantages and limitations of each paradigm.

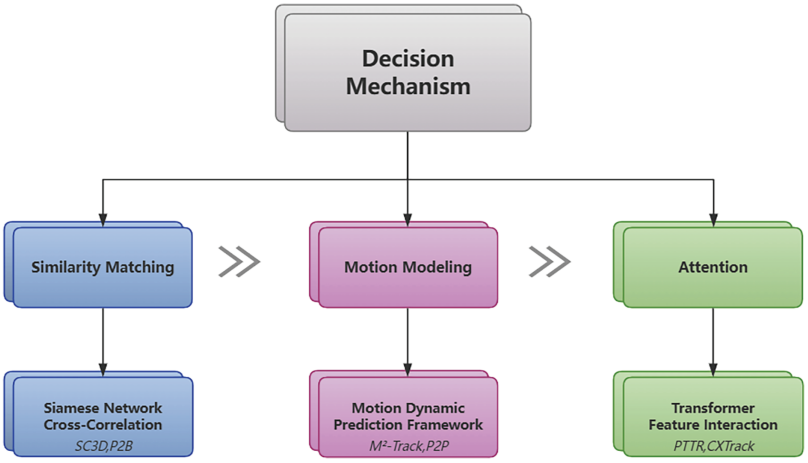


**Figure 3:** Classification of decision mechanisms.

### *3.1 Similarity Matching*

In the field of 3D object tracking, decision mechanisms based on Similarity Matching (often referred to in the literature as Siamese-based tracking or Appearance-based tracking) have long occupied a central position. Fundamentally, these methods achieve localization by modeling the feature similarity between the template object and the search region. Such approaches typically adopt a Siamese network architecture:

weight-sharing feature extraction branches process the template point cloud and search region point cloud, respectively, with efficient similarity calculation modules designed to achieve object matching [40].

As a foundational work in this direction, SC3D [41] was the first to introduce Siamese networks into the field of point cloud tracking. The method generates candidate object proposals via Kalman filtering [3], uses an encoder to map the template and candidate point clouds to a compact latent space, and then performs object matching via cosine similarity. However, its mechanism relying on iterative optimization for proposal generation not only rendered end-to-end training challenging but also introduced significant bottlenecks in computational efficiency.

#### *3.1.1 End-to-End Proposal Generation*

P2B [42] was the first to propose an end-to-end tracking framework, abandoning traditional proposal generation strategies. Its core innovation lies in a two-stage mechanism of object center localization → proposal generation: first, it localizes potential object centers in the search region via a Hough voting mechanism [5]; second, it generates object proposals via neighborhood point feature aggregation. To enhance the discriminative capability between objects and backgrounds, it designed an object-specific feature enhancement module. By performing point-wise cosine similarity calculations between template seed points and search region points, this module embeds template semantic information into the feature space of the search region, effectively enhancing object recognition capability in complex scenarios. Subsequent research, such as BAT [43], further advanced geometric prior modeling by proposing the BoxCloud representation to explicitly encode geometric relationships between points and object bounding boxes (e.g., point-to-corner distances). It also guided feature interaction between templates and search regions via a box-aware feature fusion module, significantly improving the perceptual accuracy of local object structures.

#### *3.1.2 Spatio-Temporal and Multimodal Fusion*

To tackle point cloud sparsity and object appearance variations, CAT [10] proposed a spatio-temporal context learning framework. On the spatial dimension, it dynamically expands the template range based on point cloud density via an adaptive template generation (ATG) strategy, which incorporates environmental information around the target into feature modeling. On the temporal dimension, a cross-frame aggregation (CFA) module is designed to retrieve relevant template features from historical frames via feature similarity, which fuses temporal information to enhance the robustness of object representations. MMF-Track [44] constructed a cross-modal interaction framework for geometric and visual texture features by converting RGB image texture information into pseudo-point features aligned with the point cloud space via a spatial alignment module. This method first back-projects image pixels to 3D space using intrinsic and extrinsic parameters of depth cameras, generating pseudo-point clouds with color information. It then achieves multi-scale fusion of texture and geometric features via a feature interaction module, and finally jointly optimizes bimodal similarity metrics in a similarity fusion module. GLT-T [45] achieves region-focused similarity calculation by designing a centricity loss function to weight and select seed points. This method first identifies key seed points of the target via density peak clustering, and then devises a centricity metric based on spatio-temporal motion consistency of point clouds, assigning higher weights to points in the target's core region with stable motion patterns. This dynamic weighting strategy enables similarity calculation concentrate on regions with structural stability and motion coherence, thereby significantly enhancing feature matching accuracy in occluded scenarios.

#### *3.1.3 Efficiency-Oriented Architectures*

3D-SiamRPN [46], inspired by the region proposal network (RPN) in 2D tracking [47], pioneered real-time point cloud tracking. This model devised two cross-correlation modules (point cloud-level and point-level) to efficiently integrate template and search region features and introduced a bin-based regression loss to enhance localization accuracy—thereby maintaining high precision while satisfying real-time requirements. V2B [48] introduced a voxel-to-bird's-eye-view (BEV) conversion framework tailored for sparse scenarios. It generates dense BEV feature maps via voxelization and $z$-axis max pooling, regresses object centers in 2D space to circumvent proposal quality issues induced by sparse point clouds, and further bolsters feature discriminability by generating dense target point clouds through a shape-aware learning module. GPT [49] innovatively devised a bipartite graph model, leveraging edge convolution to propagate geometric shape relationships between template points and search points. By integrating Hough voting to generate object center cues, it accomplishes 38 FPS real-time performance while sustaining high tracking accuracy.

From early exploratory efforts reliant on iterative optimization to breakthroughs in end-to-end architectures, and from single geometric feature modeling to the deep integration of spatio-temporal context and multimodal information, similarity matching methodologies have continuously evolved in tackling key challenges such as point cloud sparsity, object deformation, and real-time performance constraints. These investigations have not only propelled advancements in tracking accuracy but have also, through technological innovations such as geometric prior embedding and feature space alignment, furnished diverse solutions for 3D object tracking in complex dynamic scenarios.

### *3.2 Motion Modeling*

In contrast to appearance-based matching strategies, motion modeling effectively mitigates the adverse effects of point cloud sparsity, occlusion, and sensor noise on tracking robustness by explicitly characterizing the motion dynamics of objects in spatio-temporal sequences, thus exhibiting unique advantages in complex dynamic scenarios.

#### *3.2.1 Rigid Motion Regression*

$M^2$-Track [8] first constructed a tracking framework centered on relative motion regression. Free from the limitations of traditional appearance matching, this method reframes the tracking task as a temporal prediction problem of object motion parameters: First, it separates the target point set from point cloud sequences via a spatio-temporal joint segmentation algorithm and predicts the coarse-grained positional transformation of the target under the rigid motion assumption; then, it introduces a motion-aided shape completion module to generate a more complete 3D shape representation by fusing target point cloud information from adjacent frames, thereby refining bounding box estimation.

MTM-Tracker [50] proposed a motion-to-matching hybrid paradigm, aimed at balancing the stability of motion modeling and the flexibility of appearance matching. The method first constructs a motion dynamics model using historical frame bounding box sequences, predicts the coarse-grained position range of the target via Kalman filtering [3] to form prior constraints on the search region; then, it performs feature matching within the constrained region, achieving fine-grained object localization through joint similarity metrics of point cloud normal vectors and spatial coordinates. BEVTrack [51] projected 3D tracking tasks into the BEV space, thereby simplifying the tracking process through 2D motion regression. It first performs voxelization on the input point cloud, generates high-density BEV feature maps via max pooling along the $z$-axis, and then designs a motion regression head in the 2D space to directly predict the displacement vector and size variation of the target center.

DMT [52] presented a two-stage processing framework: "initial localization via motion prediction and refinement via MLP". Specifically, in the motion prediction stage, it leverages historical bounding box sequences to model spatio-temporal motion patterns of objects under rigid body assumptions, generating coarse-grained spatial priors to constrain the search region—thereby reducing the computational overhead of subsequent feature matching by approximately 40% compared to global search strategies. In the MLP refinement stage, the algorithm integrates local geometric descriptors of point clouds with motion cues from the prediction stage, iteratively refining localization errors through a lightweight feedforward network. This hierarchical optimization strategy not only maintains real-time performance (achieving 107 FPS on edge devices) but also enhances localization precision by 6.2% in sparse point cloud scenarios (with ≤50 points per target) compared to single-stage methods, thereby achieving a balance between tracking efficiency and localization accuracy in dynamic traffic environments.

#### *3.2.2 Part-Level and Hybrid Modeling*

P2P [53] proposed a part-level motion modeling framework that explicitly models the spatial transformation relationships of object components between adjacent frames through dual-branch feature representation in the point domain and voxel domain. Specifically, it achieves cross-frame alignment and difference fusion of component spatial structures via feature concatenation and dynamic weight assignment, enabling the model to capture subtle changes in object local motion. Experimental data shows that while maintaining a real-time inference speed of 107 FPS, it achieves a 6.7% improvement in localization accuracy compared to $M^2$-Track [8] on the KITTI [14] pedestrian tracking task, validating the critical role of local motion cues in tracking complex objects.

SiamMo [54] further explored the deep integration of Siamese network architectures with motion modeling. The model employs a dual-branch feature extractor to process point clouds from adjacent frames, respectively, captures object motion trajectories across multi-resolution scales via a spatio-temporal feature aggregation (STFA) module, and innovatively introduces a bounding box-aware feature encoding (BFE) mechanism. This mechanism embeds prior information such as object size and orientation into the motion feature space, enhancing the modeling capacity for the dynamic characteristics of rigid body motion.

From early coarse-grained position prediction based on rigid motion assumptions, to fine-grained modeling of part-level motion features, and then to the deep integration of Siamese architectures with motion dynamics, the evolution of motion modeling methods has consistently centered on the core issue of "how to more accurately characterize the spatio-temporal motion characteristics of an object". Such methods have not only overcome the constraints of point cloud data quality on tracking performance but also provided new technical pathways for addressing challenges like object occlusion, deformation, and fast motion in complex scenarios through the deep mining of temporal information.

### *3.3 Attention*

The introduction of Transformer architectures [33] marked a pivotal shift in feature modeling paradigms. By virtue of the self-attention mechanism's capability to efficiently model long-range dependencies, Transformer architectures broke through the limitations of traditional Siamese networks that rely on fixed matching modules, thereby exhibiting superior feature fusion and object localization capabilities in complex dynamic scenarios.

Note that the inter-frame attention discussed here differs from the intra-frame attention in Section 2.4. While Section 2.4 focused on intra-frame attention for extracting local geometric features within the backbone, this section addresses inter-frame attention as a decision mechanism to model the global correlation between the template and the search region.

PTTR [11], as a pioneering work in this field, first constructed a point-relation Transformer framework. It strengthened local feature interaction between the template and search region via self-attention mechanisms, and achieved global-scale feature matching and alignment through cross-attention mechanisms. This end-to-end decision paradigm abandoned the traditional approach of manually designed similarity metrics, enabling the model to adaptively learn geometric correlation features of inter-frame point clouds. PTTR++ [55] further optimized the architecture design by proposing an innovative Point-BEV dual-branch Transformer fusion strategy. This method extracts fine geometric features via MLPs in the point cloud domain, while generating dense feature maps through voxelization and max pooling in the BEV domain. It achieves complementary enhancement of point cloud details and BEV global structures via cross-branch attention mechanisms, thereby realizing multi-scale feature integration. CXTrack [56] introduced an object-centric Transformer, which propagates contextual information and object cues from historical frames to the current frame through attention, effectively mitigating feature degradation caused by object occlusion. SyncTrack [57] completely abandoned the Siamese structure and proposed a single-branch synchronous architecture. By concatenating template and search region point clouds and feeding them into the Transformer, it naturally integrates feature extraction and matching processes using a dynamic attention matrix, significantly improving accuracy while maintaining real-time performance (45 FPS).

To address the feature sampling challenge caused by point cloud sparsity, PTTR [11] proposed a relationship-aware sampling strategy. By dynamically selecting highly correlated points based on attention response values between the template and search region, it significantly reduced geometric information loss induced by random sampling. SyncTrack [57] further built upon this optimization by designing an attention-point sampling Transformer, which guides key point selection via attention weights. This approach preserves more discriminative features while maintaining computational efficiency.

Through continuous innovation in Transformer architectures, these methods have driven a paradigm shift in 3D object tracking from manual feature matching to adaptive spatiotemporal modeling. Their technological evolution not only reflects exploratory efforts in multi-scale feature fusion but also demonstrates the ongoing optimization of the balance between real-time performance and tracking accuracy.

## 4 Model Update Strategy

Model update strategies serve as a core technical component for addressing object shape variations, scene occlusion, and point cloud sparsity. As shown in Fig. 4, these mechanisms dynamically adjust tracking model parameters or feature representations to ensure the algorithm maintains stable tracking performance despite temporal variations in object appearance. The evolution of these mechanisms centers on achieving robust adaptive model updates robust adaptive model updates in unstructured point cloud data scenarios.

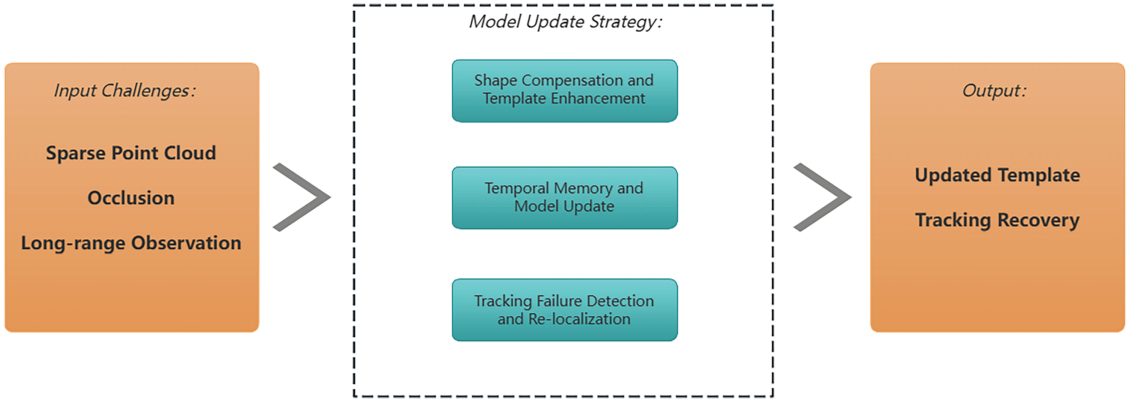


**Figure 4:** Function of model update strategy.

### *4.1 Shape Compensation and Template Enhancement*

To address key challenges such as point cloud sparsity and object occlusion, dynamic updates of tracking models are achieved through geometric structure repair and feature representation optimization. The technical core of such methods lies in addressing challenges related to object representation under sparse point cloud scenarios through approaches including point cloud geometric completion, dynamic template optimization, and virtual feature generation. Their core objective is to maintain the template's capability to precisely characterize the target's geometric properties and appearance features throughout the tracking process.

#### *4.1.1 Geometric Completion and Generation*

SC3D [41] pioneered an autoencoder-based shape completion network, which generates complete object geometry by learning the latent representation of sparse point clouds. This network constrains the similarity between generated point clouds and real shapes via Chamfer distance [58], enabling the template to maintain precise characterization of the target's geometric properties during tracking. V2B [48] further developed this approach by proposing a shape-aware feature learning module: it enhances foreground object features through a gating mechanism, generates dense point clouds by combining global-local branches, and compresses spatial information using voxelization and BEV view transformation, thereby significantly improving object localization accuracy in sparse scenarios. Such methods embed shape completion as a regularization term into the tracking framework, ensuring that the template update process balances appearance matching and geometric consistency.

#### *4.1.2 Cross-Modal and Feature Enhancement*

Facing severe point cloud loss caused by occlusion or long-distance observation, MVCTrack [59] proposed a virtual cue projection mechanism, which uses a lightweight 2D segmentation model to generate instance masks and back-projects image pixels to 3D space to create virtual point clouds. The fused input of these virtual points and original LiDAR point clouds enables the target template to retain highly discriminative features in extreme scenarios. Experiments show that this method increases the target point cloud density by 45%, particularly improving the tracking stability of small targets (e.g., pedestrians, bicycles). Boosting 3D SOT [60] leverages a target-aware projection module to map 3D point clouds to the 2D image space, leveraging pre-trained 2D trackers to extract mature matching knowledge. It subsequently achieves cross-modal knowledge transfer through IoU-weighted distillation loss, allowing the 3D tracker to

learn robust matching capabilities even with limited point cloud data. This cross-domain knowledge transfer strategy effectively mitigates the problem of insufficient training data in 3D point cloud tracking scenarios.

SCVTrack [61] proposed a synthetic object representation method based on generative adversarial networks. This approach introduces a point cloud quality assessment module to perform structural integrity detection on input sparse point clouds, and then synthesizes missing geometric surfaces using a conditional GAN [62]. Its key innovation lies in designing a multi-scale Chamfer distance and normal vector consistency joint loss function, ensuring that the synthesized point clouds not only approximate real objects in spatial position but also retain accurate surface normal information, thereby constructing high-precision target templates. PillarTrack [63] focused on robust encoding of point cloud features, proposing pyramid-encoded pillar features. By leveraging a hierarchical coordinate encoding strategy to balance numerical differences across the dimensions of point cloud coordinates, it significantly enhances the invariance of features to 3D spatial transformations. Combined with a modality-aware Transformer backbone network, this method strengthens geometric information encoding in the early stages of feature extraction and improves template discriminability in dynamic scenarios through rational allocation of computational resources. This optimization approach at the fundamental level of feature representation provides effective support for stable template updates in complex environments.

The above methods generally adopt incremental update strategies, fusing reliable point cloud information from high-confidence detection results frame by frame. Through this selective integration approach, they avoid template drift caused by single-frame error accumulation. Experimental results demonstrate that shape compensation and template enhancement mechanisms have shown significant effectiveness in challenging scenarios such as long-range tracking and complex occlusion on mainstream datasets.

### *4.2 Temporal Memory and Model Update*

Temporal memory and model update strategies have provided key technical pathways for addressing challenges such as object occlusion, deformation, and template drift in long-term tracking by exploiting spatiotemporal correlation information from historical frames. These methods rely on constructing efficient historical information management mechanisms, which enable adaptive updates of tracking models to accommodate object motion patterns through dynamic selection and fusion of historical frame features or templates. Their technological evolution has been prominently reflected in the paradigm shift from independent single-frame processing to temporal dependency modeling.

#### *4.2.1 Explicit Template Management*

TAT [64] proposed a template selection mechanism based on 3D IoU scoring. This method designed a confidence evaluation network to select high-quality templates exhibiting strong geometric matching with the current target from historical frames, thereby constructing a dynamic template set. Building on this, it enhanced cross-template feature interaction using a linear attention mechanism and performed weighted fusion of template information in chronological order via a recurrent neural network, effectively suppressing interference from low-quality early templates in current decisions. This temporal selective memory strategy enables the model to maintain tracking cues through historical high-quality templates even during temporary object occlusion, significantly improving trajectory continuity in complex interactive scenarios. MBPTrack [65] introduced an external memory bank architecture, which separately stores point cloud features and object mask information from historical frames. It processed geometric feature propagation and object semantic cue propagation separately via a decoupled Transformer module. This design avoided mutual interference between different modalities of information during fusion, achieving efficient reuse of historical object states. When the target reappears from occlusion, the complete geometric

features stored in the memory bank can quickly restore the object's representation, effectively addressing tracking interruption issues in traditional methods caused by single-frame information loss.

#### *4.2.2 Sequence and Streaming Modeling*

SeqTrack3D [66] integrated the sequence-to-sequence paradigm into 3D object tracking, constructing an end-to-end prediction framework based on historical point cloud and bounding box sequences. The method's encoder extracted multi-scale motion cues via a spatiotemporal feature pyramid, while the decoder generated target position queries conditioned on historical bounding box sequences. By leveraging self-attention mechanisms, it explicitly modeled the temporal continuity of trajectories. Its core innovation lies in transforming object motion constraints into physical law constraints of bounding box sequences, thereby compelling the network to learn position transition patterns consistent with rigid motion characteristics. This enables precise prediction of object motion trends in long-term sequence tracking. StreamTrack [67], targeting real-time requirements, designed a lightweight streaming processing architecture. It inputs only the current frame's point cloud into the backbone network and interacts with key features from historical frames through memory bank indexing. Adopting a hybrid attention mechanism, this method captures temporal changes in the target's geometric structure at the local level and models dynamic characteristics of cross-frame motion at the global level, effectively balancing computational efficiency and long-range dependency modeling capability.

Facing the challenge of tracking drift in occluded scenarios, STMD-Tracker [68] constructed a bidirectional cross-frame memory module, creatively incorporating future frame information to contribute to feature compensation for the current frame. In the forward process, the model enhanced the feature representation of the current frame using the complete object information from future frames; in the backward process, it corrected the current state through historical memory, forming a closed-loop update of spatiotemporal information. Combined with a Gaussian spatial masking mechanism, this method weights features based on the spatial distribution of target centers, effectively suppressing the influence of background interfering point clouds. It maintains precise localization of the core region even when the target is partially occluded. M3SOT [69] enhanced model robustness at the feature representation level by constructing a multi-solution space in the intermediate layers of the Transformer, simultaneously performing masked point prediction and target center localization tasks. This progressive training mechanism compelled the network to learn multi-dimensional feature representations of the target. Even with sparse point cloud inputs, it can infer the complete spatial position of the target through temporal context reasoning from historical frames, significantly improving adaptability to extremely sparse scenarios.

Essentially, these methods have continuously optimized the memory-reasoning-update closed loop in object tracking. From early rule-based template selection, to data-driven sequential modeling, and further to bidirectional compensation mechanisms integrating multimodal information, such methods have gradually overcome the limitations of single-frame processing by explicitly modeling the temporal dependency of object motion.

### *4.3 Tracking Failure Detection and Relocalization*

Addressing the issue of trajectory interruption caused by object occlusion and point cloud sparsity in complex dynamic scenarios, recent studies have significantly improved the recovery capability of tracking systems under extreme conditions by constructing precise failure detection metrics and efficient relocalization strategies.

DeepPCT [70] systematically constructed a detection-relocalization closed-loop mechanism. The core innovation of this method lies in proposing the center intersection-over-union (CIoU) metric, which extends the traditional IoU by introducing a normalized center distance. This effectively addresses the challenge of quantifying spatial relationships for non-overlapping bounding boxes. Compared to traditional metrics that only focus on region overlap, CIoU can more accurately characterize the relative positional deviation between predicted and ground truth boxes, making it particularly suitable for object localization evaluation in sparse point cloud scenarios. Upon detecting that the CIoU value falls below a predefined confidence threshold, the system automatically activates the relocalization module. This module invokes a pre-trained 3D object detector to perform high-density point cloud reconstruction and object proposal generation in the global search region, thereby achieving geometric feature matching and position correction for drifted targets. Experimental data shows that this mechanism increases the success rate by 18 percentage points in long-term tracking tasks on the KITTI [14] dataset. Notably, in extremely sparse scenarios where the number of single-object points is less than 150, the relocalization response time is improved by 40% compared to baseline methods, significantly reducing the risk of tracking failure caused by feature loss.

In the field of implicit anti-degradation research, LTTR [71] constructed a Transformer-based encoder-decoder framework, enhancing feature robustness in sparse scenarios by explicitly modeling point cloud region relationships. This method breaks through the local neighborhood limitation of traditional point cloud processing, capturing long-range dependencies between any two points using a multi-head self-attention mechanism while retaining the prior knowledge of spatial distribution of point clouds through a positional encoding module. STNet [72] and CorpNet [73] represent technical pathways for enhancing robustness via network structure optimization. STNet [72] designed an iterative cross-self-feature enhancement module: during the feature interaction phase, it embeds the global geometric information of the template object into the current frame's feature space via cross-attention, then aggregates locally semantically similar points using a self-attention mechanism, forming a multi-round progressive feature optimization process. This mechanism effectively suppresses error accumulation in short-term object occlusion scenarios, maintaining tracking trajectory continuity through iterative calibration of historical frame features. CorpNet [73] focused on multi-scale feature encoding, constructing a correlation pyramid structure with 512/256/128-point hierarchies. It simultaneously preserves high-level semantic information and low-level geometric details during feature extraction. Through a cross-layer feature fusion strategy, this method significantly enhances structural awareness capability in sparse point cloud regions, reducing the probability of tracking failure caused by local feature loss by 23%.

## 5 Experiments

### *5.1 Comparison of All Methods*

To ensure a comprehensive and rigorous evaluation of 3D single object tracking methods, we conducted systematic comparisons on two benchmark datasets: KITTI [14] and NuScenes [15], which are widely recognized for their representativeness in autonomous driving scenarios. Specifically, the KITTI dataset comprises 21 video sequences for training and 29 for testing, with each sequence containing synchronized LiDAR point clouds and camera images captured in urban, rural, and highway environments. Due to the inaccessibility of official test set labels, we further split the training set into three non-overlapping subsets (train/val/test) following standard protocols, ensuring unbiased performance validation. For NuScenes, a larger-scale dataset with richer scene diversity, it includes 1000 scenes covering various weather conditions (e.g., sunny, rainy, foggy) and time periods (day and night), which are partitioned into 700 training scenes, 150 validation scenes, and 150 test scenes to support comprehensive model generalization analysis.

In terms of evaluation categories, we focused on five typical object classes critical to autonomous driving: Car, Pedestrian, Truck, Trailer, and Bus. These categories were selected based on their high frequency in traffic scenarios and the distinct challenges they pose (e.g., small size for Pedestrians, large volume for Trucks/Trailers), ensuring the evaluation covers diverse tracking scenarios.

**Table 1:** Comparison on KITTI dataset.

| Method | Source | Car [6424] | Pedestrian [6088] | Van [1248] | Cyclist [308] | Mean [14,068] |
|---|---|---|---|---|---|---|
| SC3D [41] | CVPR19 | 41.3/57.9 | 18.2/37.8 | 40.4/47.0 | 41.5/70.4 | 31.2/48.5 |
| P2B [42] | CVPR20 | 56.2/72.8 | 28.7/49.6 | 40.8/48.4 | 32.1/44.7 | 42.4/60.0 |
| 3D-SiamRPN [46] | JSEN20 | 58.2/76.2 | 35.2/56.2 | 45.7/52.9 | 36.2/49.0 | 46.7/64.9 |
| PTT [74] | IROS21 | 67.8/81.8 | 44.9/72.0 | 43.6/52.5 | 37.2/47.3 | 55.1/74.2 |
| LTTR [71] | BMVC21 | 65.0/77.1 | 33.2/56.8 | 35.8/45.6 | 66.2/89.9 | 48.7/65.8 |
| BAT [43] | ICCV21 | 60.5/77.7 | 42.1/70.1 | 52.4/67.0 | 33.7/45.4 | 51.2/72.8 |
| V2B [48] | NIPS21 | 70.5/81.3 | 48.3/73.5 | 50.1/58.0 | 40.8/49.7 | 58.4/75.2 |
| GPT [49] | AAAI22 | 59.1/75.6 | 35.2/63.6 | 49.6/60.6 | 34.3/46.3 | 47.4/68.4 |
| STNet [72] | ECCV22 | 72.1/84.0 | 49.9/77.2 | 58.0/70.6 | 73.5/93.7 | 61.3/80.1 |
| $M^2$-Track [8] | CVPR22 | 65.5/80.8 | 61.5/88.2 | 53.8/70.7 | 73.2/93.5 | 62.9/83.4 |
| PTTR [11] | CVPR22 | 65.2/77.4 | 50.9/81.6 | 52.5/61.8 | 65.1/90.5 | 57.9/78.2 |
| TAT [64] | ACCV22 | 72.2/83.3 | 57.4/84.4 | 58.9/69.2 | 74.2/93.9 | 64.7/82.8 |
| GLT-T [45] | AAAI23 | 68.2/82.1 | 52.4/78.8 | 52.6/62.9 | 68.9/92.1 | 60.1/79.3 |
| CAT [10] | TNNLS23 | 66.6/81.8 | 51.6/77.7 | 53.1/69.8 | 67.0/90.1 | 58.9/79.1 |
| BEVTrack [51] | arXiv23 | 74.9/86.5 | 69.5/94.3 | 66.0/77.2 | 77.0/94.7 | 71.8/89.2 |
| DMT [52] | TITS23 | 66.4/79.4 | 48.1/77.9 | 53.3/65.6 | 70.4/92.6 | 55.1/75.8 |
| CorpNet [73] | CVPR23 | 73.6/84.1 | 55.6/82.4 | 58.7/66.5 | 74.3/94.2 | 64.5/82.0 |
| CXTrack [56] | CVPR23 | 69.1/81.6 | 67.0/91.5 | 60.0/71.8 | 74.2/94.3 | 67.5/85.3 |
| SyncTrack [57] | ICCV23 | 73.3/85.0 | 54.7/80.5 | 60.3/70.0 | 73.1/93.8 | 64.1/81.9 |
| MBPTrack [65] | ICCV23 | 73.4/84.8 | 68.6/93.9 | 61.3/72.7 | 76.7/94.3 | 70.3/87.9 |
| MMF-Track [44] | TIV23 | 73.9/84.1 | 61.7/89.3 | 59.3/72.5 | 75.9/95.0 | 67.4/85.6 |
| MTM-Tracker [50] | RAL23 | 73.1/84.5 | 70.4/95.1 | 60.8/74.2 | 76.7/94.6 | 70.9/88.4 |
| SCVTrack [61] | AAAI24 | 68.7/81.9 | 62.0/89.1 | 58.6/72.8 | 77.4/94.4 | – |
| StreamTrack [67] | AAAI24 | 72.6/83.7 | 70.5/94.7 | 61.0/76.9 | 78.1/94.6 | 70.8/88.1 |
| SeqTrack3D [66] | ICRA24 | – | – | – | – | – |
| PTTR++ [55] | TPAMI24 | 73.4/84.5 | 55.2/84.7 | 55.1/62.2 | 71.6/92.8 | 63.9/82.8 |
| VoxelTrack [6] | arXiv24 | 72.5/84.7 | 67.8/92.6 | 69.8/683.6 | 75.1/94.7 | 70.4/88.3 |
| MVCTrack [59] | arXiv24 | – | – | – | – | – |
| SiamMo [54] | arXiv24 | 76.3/88.1 | 68.6/93.9 | 67.9/80.5 | 78.5/94.8 | 72.3/90.1 |
| M3SOT [69] | AAAI24 | 75.9/87.4 | 66.6/92.5 | 59.4/74.7 | 70.3/93.4 | 70.3/88.6 |
| MemDisst [60] | ECCV24 | 74.1/85.6 | 69.1/94.1 | 66.6/79.3 | 77.2/94.7 | 71.3/88.9 |
| PillarTrack [63] | arXiv24 | 74.2/85.1 | 59.7/84.7 | 61.0/69.2 | 78.0/95.0 | 66.8/83.7 |
| STMD-Tracker [68] | arXiv24 | 73.7/85.2 | 68.9/94.2 | 61.4/72.7 | 76.9/94.9 | 70.6/88.2 |
| P2P [53] | IJCV25 | 73.6/85.7 | 69.6/94.0 | 70.3/83.9 | 75.5/94.6 | 71.7/89.4 |

Note: The best and second-best results are highlighted in **red** and **blue**, respectively. "-" denotes that the result is not available.

The quantitative evaluation results of state-of-the-art algorithms on the KITTI and NuScenes datasets are presented in Tables 1 and 2, respectively ("-" indicates that there are no relevant experiments for this method; the best and second-best results are highlighted in red and blue, respectively). To ensure a fair comparison, we adopt Success and Precision defined in one-pass evaluation (OPE) [75,76] as the evaluation metrics. Success denotes the Area Under the Curve (AUC) for the plot of the ratio of frames in which the IoU between the predicted box and the ground truth box exceeds a threshold (ranging from 0 to 1). Meanwhile, Precision denotes the AUC for the plot of the ratio of frames in which the distance between their centers (i.e., the predicted box and the ground truth box) is within a threshold (ranging from 0 to 2 m).

**Table 2:** Comparison on NuScenes dataset.

| Method | Source | Car [64,159] | Pedestrian [33,227] | Truck [13,587] | Trailer [3352] | Bus [2953] | Mean [117,278] |
|---|---|---|---|---|---|---|---|
| SC3D [41] | CVPR19 | 22.31/21.93 | 11.29/12.65 | 30.67/27.73 | 35.28/28.12 | 29.35/24.08 | 20.70/20.20 |
| P2B [42] | CVPR20 | 38.81/43.18 | 28.39/52.24 | 42.95/41.59 | 48.96/40.05 | 32.95/27.41 | 36.48/45.08 |
| 3D-SiamRPN [46] | JSEN20 | – | – | – | – | – | – |
| PTT [74] | IROS21 | 41.22/45.26 | 19.33/32.03 | 50.23/48.56 | 51.70/46.50 | 39.40/36.70 | 36.33/41.72 |
| LTTR [71] | BMVC21 | – | – | – | – | – | – |
| BAT [43] | ICCV21 | 40.73/43.29 | 28.83/53.32 | 45.34/42.58 | 52.59/44.89 | 35.44/28.01 | 38.10/45.71 |
| V2B [48] | NIPS21 | 54.40/59.70 | 30.10/55.40 | 53.70/54.50 | 54.90/51.44 | – | – |
| GPT [49] | AAAI22 | – | – | – | – | – | – |
| STNet [72] | ECCV22 | – | – | – | – | – | – |
| $M^2$-Track [8] | CVPR22 | 55.85/65.09 | 32.10/60.92 | 57.36/59.54 | 57.61/58.26 | 51.39/51.44 | 49.23/62.73 |
| PTTR [11] | CVPR22 | 51.89/58.61 | 29.90/45.09 | 45.30/44.74 | 45.87/38.36 | 43.14/37.74 | 44.50/52.07 |
| TAT [64] | ACCV22 | – | – | – | – | – | – |
| GLT-T [45] | AAAI23 | 48.52/54.29 | 31.74/56.49 | 52.74/51.43 | 57.60/52.01 | 44.55/40.69 | 44.42/54.33 |
| CAT [10] | TNNLS23 | 43.34/49.41 | 30.68/56.67 | 47.64/48.10 | 57.90/55.31 | 43.30/41.42 | 40.67/51.28 |
| BEVTrack [51] | arXiv23 | 64.31/71.14 | 46.28/76.77 | 66.83/67.04 | 75.54/71.62 | 61.09/56.68 | 59.71/71.19 |
| DMT [52] | TITS23 | – | – | – | – | – | – |
| CorpNet [73] | CVPR23 | – | – | – | – | – | – |
| CXTrack [56] | CVPR23 | 48.92/55.61 | 31.67/56.64 | 51.40/50.93 | 60.64/54.44 | 40.11/35.83 | 44.43/54.83 |
| SyncTrack [57] | ICCV23 | – | – | – | – | – | – |
| MBPTrack [65] | ICCV23 | 62.47/70.41 | 45.32/74.03 | 62.18/63.31 | 65.14/61.33 | 55.41/51.76 | 57.48/69.88 |
| MMF-Track [44] | TIV23 | 50.73/58.85 | 32.80/66.25 | – | – | 52.73/52.98 | – |
| MTM-Tracker [50] | RAL23 | 57.05/65.94 | 37.08/72.80 | 59.37/63.69 | 59.73/60.44 | 55.46/54.31 | 51.70/67.17 |
| SCVTrack [61] | AAAI24 | 58.9/67.7 | 34.5/61.5 | 60.6/61.4 | 59.5/60.1 | 54.3/53.6 | 52.1/64.7 |
| StreamTrack [67] | AAAI24 | 62.65/70.81 | 38.43/68.58 | 64.67/66.60 | 66.67/64.27 | 60.66/59.74 | 55.75/69.22 |
| SeqTrack3D [66] | ICRA24 | 62.55/71.46 | 39.94/68.57 | 60.97/63.04 | 68.37/61.76 | 54.33/53.52 | 55.92/68.94 |
| PTTR++ [55] | TPAMI24 | 59.96/66.73 | 32.49/50.50 | 59.85/61.20 | 54.51/50.28 | 53.98/51.22 | 51.86/60.63 |
| VoxelTrack [6] | arXiv24 | 63.9/71.6 | 46.8/75.9 | 64.8/65.9 | 69.5/64.3 | 60.1/57.7 | 59.0/71.4 |
| MVCTrack [59] | arXiv24 | 66.76/73.76 | 47.34/76.64 | 66.21/66.33 | 72.73/69.80 | 60.20/58.88 | 61.20/73.22 |
| SiamMo [54] | arXiv24 | 64.95/72.24 | 46.23/76.25 | 68.22/68.81 | 74.21/70.63 | 65.63/62.07 | 60.31/72.68 |
| M3SOT [69] | AAAI24 | – | – | – | – | – | – |
| MemDisst [60] | ECCV24 | – | – | – | – | – | – |
| PillarTrack [63] | arXiv24 | 47.12/57.72 | 34.18/64.93 | 54.82/54.41 | 57.70/54.63 | 44.68/40.73 | 44.59/58.86 |
| STMD-Tracker [68] | arXiv24 | 63.05/71.24 | 46.86/75.27 | 62.87/63.96 | 65.24/62.03 | 56.02/52.88 | 58.33/70.81 |
| P2P [53] | IJCV25 | 65.15/72.90 | 46.43/75.08 | 64.96/65.96 | 70.46/66.86 | 59.02/56.56 | 59.84/72.13 |

Note: The best and second-best results are highlighted in **red** and **blue**, respectively. "-" denotes that the result is not available.

Table 1 summarizes SOTA 3D single-object tracking performance on the KITTI dataset, with evaluations based on Success and Precision. SiamMo [54] establishes a new SOTA with a mean Success of 72.3 and a mean Precision of 90.1, outperforming MTM-Tracker [50] (70.9/88.4) by 1.4 points. This performance gain stems from its spatio-temporal feature aggregation (STFA) module, which explicitly models cross-frame

motion trajectories, which is a critical capability for mitigating partial occlusion and fast motion in urban and highway scenarios.

Table 2 presents performance evaluations on NuScenes—a larger-scale dataset featuring diverse weather conditions (e.g., rain, fog) and sparse long-range point clouds (>50 m), which amplifies generalization challenges. MVCTrack [59] establishes itself as the SOTA (61.20 in mean Success, 73.22 in mean Precision), outperforming SiamMo [54] (60.31/72.68) and P2P [53] (59.84/72.13). Its multi-view cross-modal fusion mechanism integrates RGB texture and LiDAR geometric features, compensating for sparse or noisy points—a critical capability for addressing NuScenes' variability.

While Tables 1 and 2 provide exhaustive details, it is beneficial to summarize the performance ceilings of different technical routes. Table 3 presents a concise comparison of the representative SOTA methods from each category. As observed, Sequence & Update methods (e.g., M3SOT [69], MVCTrack [59]) currently achieve the highest performance on both datasets, validating the importance of temporal context in handling occlusion. Motion Modeling methods (e.g., P2P [53]) show exceptional robustness on the NuScenes dataset, narrowing the gap with sequence-based methods, whereas traditional Similarity Matching approaches generally lag behind in complex sparse scenarios.

**Table 3:** Concise summary of SOTA performance by category.

| Category | Representative SOTA | KITTI (Mean) | NuScenes (Mean) | Key Characteristics |
|---|---|---|---|---|
| **Similarity Matching** | MMF-Track [44]/ VoxelTrack [6] | 67.4 | 59.0 | Robust texture fusion; Fast inference |
| **Motion Modeling** | P2P [53] | 71.7 | 59.8 | Explicit kinematic constraints; Rigid-body focus |
| **Transformer** | CorpNet [73]/ PTTR++ [55] | 64.5 | 51.9 | Long-range dependency; High computation |
| **Sequence & Update** | M3SOT [69]/ MVCTrack [59] | **72.3** | **61.2** | Temporal consistency; Best occlusion handling |

Regarding backbone impact, Transformer-based backbones consistently outperform (e.g., PTTR [11]) consistently outperform PointNet++ [20] based ones by approximately 5% in Success, validating the importance of global context capture in sparse point clouds. In terms of decision mechanism effectiveness, Motion-centric methods (e.g., $M^2$-Track [8]) show superior robustness on rigid objects like cars compared to Similarity Matching, as they explicitly utilize kinematic constraints. However, for non-rigid objects such as pedestrians, attention-based methods show better adaptability. Finally, concerning update strategies, methods utilizing Temporal Memory (e.g., MBPTrack [65]) demonstrate a significant performance boost in long-term tracking scenarios compared to single-frame baselines.

### *5.2 Visualization*

We select BAT [43] as the baseline for visualization because it represents the foundational "Box-Aware" paradigm and serves as a standard open-source benchmark for validating recent advancements.

To mitigate the paucity of visual evaluations in existing 3D point cloud tracking literature, we integrate visualization data from M$^2$-Track [8] and compare it with BAT [43], as illustrated in Figs. 5 and 6. These visual comparisons offer mechanistic insights into the representative tracking sequence, thereby facilitating a nuanced understanding of the efficacy of motion modeling and trajectory estimation in 3D object tracking.

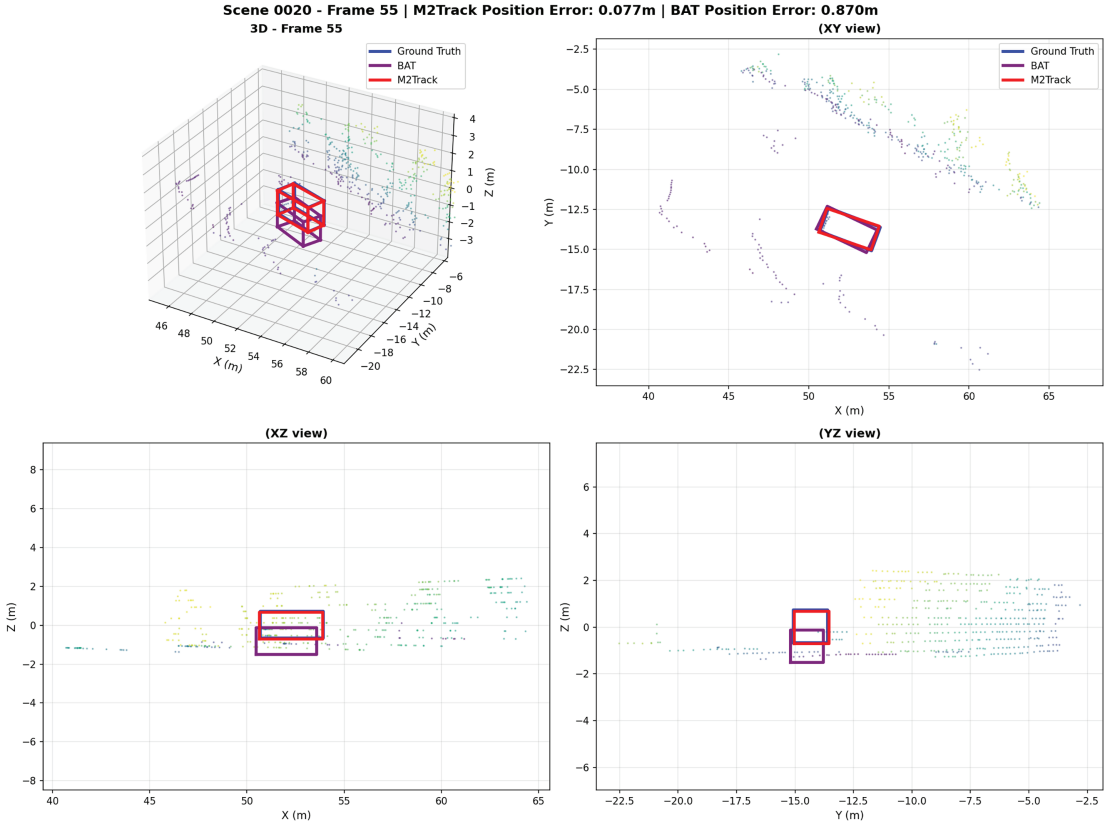


**Figure 5:** Visualization of tracking results.

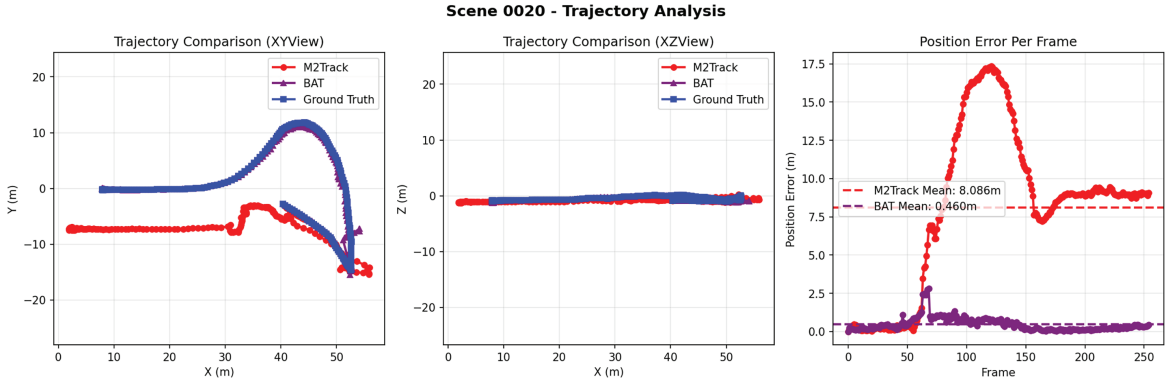


**Figure 6:** Trajectory analysis: long-term consistency and per-frame position error.

In Scene 0020 (Frame 55), the notable position error of M$^2$-Track (0.077 m vs. BAT's 0.870 m in single-frame localization) exposes limitations in its motion modeling under fine-grained spatial constraints. The discrepancy between M$^2$-Track's bounding box and the ground truth across multi-view projections (e.g., XY,

XZ views) indicates that its motion-driven localization strategy grapples with precise spatial alignment in dense point cloud clusters. Conversely, BAT exhibits superior single-frame localization accuracy (0.870 m error) and trajectory consistency, attributed to its effective fusion of geometric features and motion priors—underscoring the pivotal role of multi-modal information integration in precise tracking. Notably, BAT's consistent alignment with the ground truth in both single-frame and trajectory analyses corroborates that fine-grained feature matching, coupled with robust motion regularization, constitutes an effective approach to enhancing tracking precision and temporal stability. This design aligns with methodologies emphasizing spatiotemporal coherence and multi-view geometric consistency, collectively indicating an emerging trend of integrating precise spatial alignment with temporal continuity in 3D tracking.

In the trajectory analysis of Scene 0020, $M^2$-Track's substantial mean position error (8.086 m) and trajectory divergence highlight the inadequacy of its motion-centric modeling in sustaining long-term tracking consistency under complex spatial dynamics. This shortcoming stands in contrast to BAT's stable trajectory (mean error 0.460 m), which benefits from adaptive motion correction and multi-frame feature aggregation—mechanisms that bolster robustness by explicitly modeling spatiotemporal dependencies. Furthermore, BAT's precision in both single-frame localization and trajectory estimation aligns with a convergent technical trajectory of methodologies emphasizing the tight integration of spatial geometry and temporal motion, collectively propelling tracking models toward spatiotemporal co-optimization.

These visual findings not only validate the efficacy of quantitative metrics but also unveil two core trends in contemporary 3D tracking: a paradigm shift from the pursuit of single-frame precision to the assurance of long-term trajectory consistency, and a methodological evolution from motion-only or feature-only modeling to integrated spatiotemporal-feature fusion. As paradigmatic examples, BAT's multi-view alignment mechanism and $M^2$-Track's motion modeling strategy demonstrate that precise spatial alignment and robust motion regularization are pivotal strategies for addressing long-term trajectory divergence and fine-grained localization errors in 3D tracking.

## 6 Discussion

### *6.1 Technological Evolution*

The field has witnessed a paradigm shift from 2D-inspired Similarity Matching to 3D Motion Modeling, and recently to Sequence-based learning. This evolution reflects a growing capability to handle the intrinsic disorder and sparsity of point clouds, moving from simple texture-less matching to sophisticated dynamic state estimation.

### *6.2 Limitations and Open Challenges*

3D SOT has witnessed substantial advancements, yet critical challenges persist, limiting its robustness in real-world scenarios. Extreme point cloud sparsity remains a primary bottleneck: long-range targets (e.g., pedestrians 100 m away with 20–50 points) or sensor noise in adverse weather (60% density reduction in rain/fog) degrade feature stability, leading to a 30% higher drift rate in state-of-the-art methods compared to dense scenarios. This sparsity exacerbates mismatches in similarity matching frameworks and propagates errors in motion modeling pipelines.

Long-term occlusion further disrupts tracking continuity. Motion-centric methods relying on local inter-frame cues fail to capture non-rigid dynamics or complex trajectories, while similarity-based approaches lose discriminative features, resulting in irreversible drift. Non-rigid deformation of targets (e.g., pedestrian limb movement) challenges rigid-body assumptions, with part-level models still lacking dense correspondence to model fine-grained changes.

The accuracy-efficiency trade-off remains unresolved. Transformer-based architectures achieve superior performance but incur quadratic computational complexity with respect to point count, hindering real-time deployment on edge devices. Lightweight alternatives optimize speed but sacrifice precision in sparse scenes. Additionally, limited generalization across diverse scenarios (urban vs. off-road, varying sensor quality) persists, as models overfit to dataset-specific patterns rather than generalizable geometric or motion priors.

## 7 Conclusion

In this survey, we have presented a systematic and comprehensive review of 3D single object tracking in point clouds. We established a unified taxonomy covering feature extraction backbones, decision mechanisms, and model update strategies, and quantitatively evaluated state-of-the-art methods across the KITTI and NuScenes benchmarks. Our analysis reveals that while significant progress has been made in handling unstructured data, the field still faces bottlenecks in extreme sparsity and long-term robustness. Drawing from these insights, we conclude by outlining key directions for the next generation of tracking systems.

Future research must prioritize: multimodal fusion to mitigate sparsity; lightweight architectures for edge deployment; and physics-informed motion modeling. Furthermore, the development of trustworthy and explainable tracking systems is imperative. Current models often act as black boxes, limiting their adoption in safety-critical domains. Future works should integrate counterfactual reasoning to diagnose tracking failures and drifts by analyzing how minimal perturbations in point clouds affect tracking outcomes. As demonstrated in recent studies on static perception [77], explainable perturbation-based analysis can reveal model sensitivities, and extending this to dynamic tracking is a promising avenue for expert-guided reliability. These advancements will accelerate 3D SOT deployment in autonomous driving, robotics, and intelligent surveillance, enabling more reliable environmental perception in dynamic real-world scenarios.

**Acknowledgement:** None.

**Funding Statement:** This work was supported by the National Natural Science Foundation of China (Nos. 62306049, 92471207 and W2421089), the General Program of Chongqing Natural Science Foundation (No. CSTB2023NSCQ-MSX0665), and the Fundamental Research Funds for the Central Universities (No. 2024CDJXY008).

**Author Contributions:** The authors confirm contribution to the paper as follows: Conceptualization, Bo Huang and Yihao Kuang; methodology, Yihao Kuang and Hong Zhang; writing—original draft preparation, Yihao Kuang, Jiaqi Wang and Lingyu Jin; writing—review and editing, Yihao Kuang and Bo Huang. All authors reviewed and approved the final version of the manuscript.

**Availability of Data and Materials:** Not applicable.

**Ethics Approval:** Not applicable.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Table of Abbreviations

| Abbreviation | Full Term |
|---|---|
| SOT | Single Object Tracking |
| BEV | Bird's Eye View |
| SSM | State Space Model |
| ATG | Adaptive Template Generation |
| CFA | Cross-Frame Aggregation |

| | |
|---|---|
| STFA | Spatio-Temporal Feature Aggregation |
| BFE | Bounding Box-Aware Feature Encoding |
| IoU/CIoU | Intersection over Union/Center IoU |
| FPS | Frames Per Second |
| RPN | Region Proposal Network |
| MLP | Multi-Layer Perceptron |
| AUC | Area Under the Curve |
| SOTA | State-of-the-Art |

## References


1. Hinterstoisser S, Lepetit V, Ilic S, Holzer S, Bradski G, Konolige K, et al. Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes. In: Asian Conference on Computer Vision. Cham, Switzerland: Springer; 2012. p. 548–62. doi:10.1007/978-3-642-37331-2_42.
2. Henriques JF, Caseiro R, Martins P, Batista J. High-speed tracking with kernelized correlation filters. IEEE Trans Pattern Anal Mach Intell. 2014;37(3):583–96. doi:10.1109/TPAMI.2014.2345390.
3. Kalman RE. A new approach to linear filtering and prediction problems. J Basic Eng. 1960;82(1):35–45. doi:10.1115/1.3662552.
4. 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. Piscataway, NJ, USA: IEEE; 2021. p. 11784–93. doi:10.1109/CVPR46437.2021.01161.
5. Qi CR, Litany O, He K, Guibas LJ. Deep hough voting for 3D object detection in point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. Piscataway, NJ, USA: IEEE; 2019. p. 9277–86. doi:10.1109/ICCV.2019.00937.
6. Lu Y, Nie J, He Z, Gu H, Lv X. VoxelTrack: exploring voxel representation for 3D point cloud object tracking. arXiv:2408.02263. 2024. doi:10.48550/arxiv.2408.02263.
7. Huang Z, Wen Y, Wang Z, Ren J, Jia K. Surface reconstruction from point clouds: a survey and a benchmark. IEEE Trans Pattern Anal Mach Intell. 2024;46(10):6762–83. doi:10.1109/TPAMI.2024.3429209.
8. Zheng C, Yan X, Zhang H, Wang B, Cheng S, Cui S, et al. Beyond 3D siamese tracking: a motion-centric paradigm for 3D single object tracking in point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2022. p. 8111–20. doi:10.48550/arXiv.2203.01730.
9. Ku J, Mozifian M, Lee J, Harakeh A, Waslander SL. Joint 3D proposal generation and object detection from view aggregation. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Piscataway, NJ, USA: IEEE; 2018. p. 1–8. doi:10.1109/IROS.2018.8594049.
10. Gao J, Yan X, Zhao W, Lyu Z, Liao Y, Zheng C. Spatio-temporal contextual learning for single object tracking on point clouds. IEEE Trans Neural Netw Learn Syst. 2024;35(4):4754–66. doi:10.1109/TNNLS.2022.3233562.
11. Zhou C, Luo Z, Luo Y, Liu T, Pan L, Cai Z, et al. PTTR: relational 3D point cloud object tracking with transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2022. p. 8531–40. doi:10.48550/arXiv.2112.02857.
12. Gu L, Wei M, Yan X, Zhu D, Zhao W, Xie H, et al. YOLOO: you only learn from others once. arXiv:2409.00618. 2024. doi:10.48550/arxiv.2409.00618.
13. Hodaň T, Matas J, Obdržálek Š. On evaluation of 6D object pose estimation. In: Hua G, Jégou H, editors. Computer Vision – ECCV 2016 Workshops. Cham, Switzerland: Springer International Publishing; 2016. p. 606–19. doi:10.1007/978-3-319-49409-8_52.
14. 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. Piscataway, NJ, USA: IEEE; 2012. p. 3354–61. doi:10.1109/CVPR.2012.6248074.
15. Luo C, Yang X, Yuille A. Exploring simple 3D multi-object tracking for autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). Piscataway, NJ, USA: IEEE; 2021. p. 10488–97. doi:10.1109/ICCV48922.2021.01034.

16. Asvadi A, Girao P, Peixoto P, Nunes U. 3D object tracking using RGB and LiDAR data. In: 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC). Piscataway, NJ, USA: IEEE; 2016. p. 1255–60. doi:10.1109/ITSC.2016.7795718.
17. Wu Z, Song S, Khosla A, Yu F, Zhang L, Tang X, et al. 3D ShapeNets: a deep representation for volumetric shapes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2015. p. 1912–20. doi:10.1109/CVPR.2015.7298801.
18. Guo Y, Wang H, Hu Q, Liu H, Liu L, Bennamoun M. Deep learning for 3D point clouds: a survey. IEEE Trans Pattern Anal Mach Intell. 2020;43(12):4338–64. doi:10.1109/tpami.2020.3005434.
19. Maturana D, Scherer S. VoxNet: a 3D convolutional neural network for real-time object recognition. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Piscataway, NJ, USA: IEEE; 2015. p. 922–8. doi:10.1109/IROS.2015.7353481.
20. Qi CR, Su H, Mo K, Guibas LJ. Pointnet: deep learning on point sets for 3D classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2017. p. 652–60. doi:10.1109/CVPR.2017.16.
21. Qi CR, Yi L, Su H, Guibas LJ. Pointnet++: deep hierarchical feature learning on point sets in a metric space. arXiv:1706.02413. 2017.
22. Li Y, Bu R, Sun M, Wu W, Di X, Chen B. PointCNN: convolution on x-transformed points. arXiv:1801.07791. 2018.
23. Liu Y, Fan B, Xiang S, Pan C. Relation-shape convolutional neural network for point cloud analysis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2019. p. 8895–904. doi:10.1109/CVPR.2019.00910.
24. Zhao H, Jiang L, Fu CW, Jia J. Pointweb: enhancing local neighborhood features for point cloud processing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2019. p. 5565–73. doi:10.1109/CVPR.2019.00571.
25. Yang Y, Feng C, Shen Y, Tian D. Foldingnet: point cloud auto-encoder via deep grid deformation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2018. p. 206–15. doi:10.1109/CVPR.2018.00029.
26. Wu W, Qi Z, Li F. Deep convolutional networks on 3D point clouds. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE; 2019. p. 9613–22. doi:10.1109/CVPR.2019.00985.
27. Xu M, Ding R, Zhao H, Qi X. Paconv: position adaptive convolution with dynamic kernel assembling on point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2021. p. 3173–82. doi:10.48550/arXiv.2103.14635.
28. Wang Y, Sun Y, Liu Z, Sarma SE, Bronstein MM, Solomon JM. Dynamic graph CNN for learning on point clouds. ACM Trans Graph. 2019;38(5):1–12. doi:10.1145/3326362.
29. Jiang B, Zhang Z, Lin D, Tang J, Luo B. Semi-supervised learning with graph learning-convolutional networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2019. p. 11313–20. doi:10.1109/CVPR.2019.01157.
30. Xu Y, Fan T, Xu M, Zeng L, Qiao Y. Spidercnn: deep learning on point sets with parameterized convolutional filters. In: Proceedings of the European Conference on Computer Vision (ECCV). Cham, Switzerland: Springer; 2018. p. 87–102. doi:10.1007/978-3-030-01237-3_6.
31. Shi W, Rajkumar R. Point-GNN: graph neural network for 3D object detection in a point cloud. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2020. p. 1711–9. doi:10.1109/CVPR42600.2020.00178.
32. Yu X, Tang L, Rao Y, Huang T, Zhou J, Lu J. Point-BERT: pre-training 3D point cloud transformers with masked point modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2022. p. 19313–22. doi:10.48550/arXiv.2111.14819.
33. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. In: Advances in neural information processing systems. Vol. 30. Red Hook, NY, USA: Curran Associates, Inc.; 2017 p. 5998–6008. doi:10.65215/ctdc8e75.

34. Guo MH, Cai JX, Liu ZN, Mu TJ, Martin RR, Hu SM. PCT: point cloud transformer. Comput Vis Media. 2021;7(2):187–99. doi:10.1007/s41095-021-0229-5.
35. Yang YQ, Guo YX, Xiong JY, Liu Y, Pan H, Wang PS, et al. Swin3D: a pretrained transformer backbone for 3D indoor scene understanding. Comput Vis Media. 2025;11(1):83–101. doi:10.26599/CVM.2025.9450383.
36. Zhao H, Jiang L, Jia J, Torr PH, Koltun V. Point transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. Piscataway, NJ, USA: IEEE; 2021. p. 16259–68. doi:10.1109/ICCV48922.2021.01595.
37. Liu X, Han Z, Liu YS, Zwicker M. Point2sequence: learning the shape representation of 3D point clouds with an attention-based sequence to sequence network. In: Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto, CA, USA: AAAI Press; 2019. p. 8778–85. doi:10.1609/aaai.v33i01.33018778.
38. Liang D, Zhou X, Xu W, Zhu X, Zou Z, Ye X, et al. Pointmamba: a simple state space model for point cloud analysis. arXiv:2402.10739. 2024. doi:10.48550/arxiv.2402.10739.
39. Yang N, Wang Y, Liu Z, Li M, An Y, Zhao X. SMamba: sparse mamba for event-based object detection. In: Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto, CA, USA: AAAI Press; 2025. p. 9229–37.
40. Bertinetto L, Valmadre J, Henriques JF, Vedaldi A, Torr PH. Fully-convolutional siamese networks for object tracking. In: European Conference on Computer Vision. Cham, Switzerland: Springer; 2016. p. 850–65. doi:10.1007/978-3-319-48881-3_56.
41. Giancola S, Zarzar J, Ghanem B. Leveraging shape completion for 3D siamese tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2019. p. 1359–68. doi:10.1109/CVPR.2019.00145.
42. Qi H, Feng C, Cao Z, Zhao F, Xiao Y. P2B: point-to-box network for 3D object tracking in point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2020. p. 6329–38. doi:10.1109/CVPR42600.2020.00636.
43. Zheng C, Yan X, Gao J, Zhao W, Zhang W, Li Z, et al. Box-aware feature enhancement for single object tracking on point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. Piscataway, NJ, USA: IEEE; 2021. p. 13199–208. doi:10.48550/arXiv.2108.04728.
44. Li Z, Cui Y, Lin Y, Fang Z. Mmf-track: multi-modal multi-level fusion for 3D single object tracking. IEEE Trans Intell Veh. 2023;9(1):1817–29. doi:10.1109/tiv.2023.3326790.
45. Nie J, He Z, Yang Y, Gao M, Zhang J. GLT-T: global-local transformer voting for 3D single object tracking in point clouds. In: Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto, CA, USA: AAAI Press; 2023. p. 1957–65. doi:10.1609/aaai.v37i2.25287.
46. Fang Z, Zhou S, Cui Y, Scherer S. 3D-siamrpn: an end-to-end learning method for real-time 3D single object tracking using raw point cloud. IEEE Sens J. 2020;21(4):4995–5011. doi:10.1109/JSEN.2020.3033034.
47. Li B, Yan J, Wu W, Zhu Z, Hu X. High performance visual tracking with siamese region proposal network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2018. p. 8971–80. doi:10.1109/CVPR.2018.00935.
48. Hui L, Wang L, Cheng M, Xie J, Yang J. 3D siamese voxel-to-bev tracker for sparse point clouds. Adv Neural Inf Process Syst. 2021;34:28714–27. doi:10.48550/arXiv.2111.04426.
49. Park M, Seong H, Jang W, Kim E. Graph-based point tracker for 3D object tracking in point clouds. In: Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto, CA, USA: AAAI Press; 2022. p. 2053–61. doi:10.1609/aaai.v36i2.20101.
50. Li Z, Lin Y, Cui Y, Li S, Fang Z. Motion-to-matching: a mixed paradigm for 3D single object tracking. IEEE Robot Autom Lett. 2023;9(2):1468–75. doi:10.1109/LRA.2023.3347143.
51. Yang Y, Deng Y, Fan J, Zhang J, Zha ZJ. Bevtrack: a simple and strong baseline for 3D single object tracking in bird's-eye view. In: Proceedings of the 31st ACM International Conference on Multimedia. New York, NY, USA: ACM; 2023. p. 1819–28.
52. Xia Y, Wu Q, Li W, Chan AB, Stilla U. A lightweight and detector-free 3D single object tracker on point clouds. IEEE Trans Intell Transp Syst. 2023;24(5):5543–54. doi:10.48550/arXiv.2203.04232.

53. Nie J, Xie F, Zhou S, Zhou X, Chae DK, He Z. P2P: part-to-part motion cues guide a strong tracking framework for LiDAR point clouds. Int J Comput Vis. 2025;133(8):5326–42. doi:10.1007/s11263-025-02430-6.
54. Yang Y, Deng Y, Zhang J, Gu H, Dong Z. SiamMo: siamese motion-centric 3D object tracking. arXiv:2408.01688. 2024. doi:10.48550/arxiv.2408.01688.
55. Luo Z, Zhou C, Pan L, Zhang G, Liu T, Luo Y, et al. Exploring point-bev fusion for 3D point cloud object tracking with transformer. IEEE Trans Pattern Anal Mach Intell. 2024;46(9):5921–35. doi:10.1109/TPAMI.2024.3373693.
56. Xu TX, Guo YC, Lai YK, Zhang SH. CXTrack: improving 3D point cloud tracking with contextual information. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2023. p. 1084–93. doi:10.1109/CVPR52729.2023.00111.
57. Ma T, Wang M, Xiao J, Wu H, Liu Y. Synchronize feature extracting and matching: a single branch framework for 3D object tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. Piscataway, NJ, USA: IEEE; 2023. p. 9953–63. doi:10.1109/iccv51070.2023.00913.
58. Fan H, Su H, Guibas LJ. A point set generation network for 3D object reconstruction from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2017. p. 605–13. doi:10.1109/CVPR.2017.264.
59. Hu Z, Zhou S, Zhao S, Yuan Z, Liang CJ. MVCTrack: boosting 3D point cloud tracking via multimodal-guided virtual cues. arXiv:2412.02734. 2024. doi:10.48550/arxiv.2412.02734.
60. Wu Q, Xia Y, Wan J, Chan AB. Boosting 3D single object tracking with 2D matching distillation and 3D pre-training. In: European Conference on Computer Vision. Cham, Switzerland: Springer; 2024. p. 270–88. doi:10.1007/978-3-031-73254-6_16.
61. Zhang J, Zhou Z, Lu G, Tian J, Pei W. Robust 3D tracking with quality-aware shape completion. In: Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto, CA, USA: AAAI Press; 2024. p. 7160–8. doi:10.1609/aaai.v38i7.28544.
62. Mirza M, Osindero S. Conditional generative adversarial nets. arXiv:1411.1784. 2014. doi:10.48550/arxiv.1411.1784.
63. Xu W, Zhou S, Yuan Z. PillarTrack: redesigning pillar-based transformer network for single object tracking on point clouds. arXiv:2404.07495. 2024. doi:10.48550/arxiv.2404.07495.
64. Lan K, Jiang H, Xie J. Temporal-aware siamese tracker: integrate temporal context for 3D object tracking. In: Proceedings of the Asian Conference on Computer Vision. Cham, Switzerland: Springer; 2022. p. 399–414.
65. Xu TX, Guo YC, Lai YK, Zhang SH. Mbptrack: improving 3D point cloud tracking with memory networks and box priors. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. Piscataway, NJ, USA: IEEE; 2023. p. 9911–20. doi:10.1109/ICCV51070.2023.00909.
66. Lin Y, Li Z, Cui Y, Fang Z. SeqTrack3D: exploring sequence information for robust 3D point cloud tracking. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). Piscataway, NJ, USA: IEEE; 2024. p. 6959–65. doi:10.1109/ICRA57147.2024.10611238.
67. Luo Z, Zhang G, Zhou C, Wu Z, Tao Q, Lu L, et al. Modeling continuous motion for 3D point cloud object tracking. In: Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto, CA, USA: AAAI Press; 2024. p. 4026–34. doi:10.48550/arXiv.2303.07605.
68. Sun S, Wang C, Liu X, Shi C, Ding Y, Xi G. Spatio-temporal bi-directional cross-frame memory for distractor filtering point cloud single object tracking. arXiv:2403.15831. 2024. doi:10.48550/arxiv.2403.15831.
69. Liu J, Wu Y, Gong M, Miao Q, Ma W, Xu C, et al. M3SOT: multi-frame, multi-field, multi-space 3D single object tracking. In: Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto, CA, USA: AAAI Press; 2024. p. 3630–8. doi:10.1609/aaai.v38i4.28152.
70. Liu H, Hu Q, Ma Y, Xie K, Guo Y. DeepPCT: single object tracking in dynamic point cloud sequences. IEEE Trans Instrum Meas. 2022;72:1–12. doi:10.1109/TIM.2022.3232092.
71. Cui Y, Fang Z, Shan J, Gu Z, Zhou S. 3D object tracking with transformer. arXiv:2110.14921. 2021. doi:10.48550/arXiv.2110.14921.
72. Hui L, Wang L, Tang L, Lan K, Xie J, Yang J. 3D siamese transformer network for single object tracking on point clouds. In: European Conference on Computer Vision. Cham, Switzerland: Springer; 2022. p. 293–310. doi:10.48550/arXiv.2207.11995.

73. Wang M, Ma T, Zuo X, Lv J, Liu Y. Correlation pyramid network for 3D single object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2023. p. 3216–25. doi:10.48550/arXiv.2305.09195.
74. Shan J, Zhou S, Fang Z, Cui Y. PTT: point-track-transformer module for 3D single object tracking in point clouds. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Piscataway, NJ, USA: IEEE; 2021. p. 1310–6. doi:10.48550/arXiv.2108.06455.
75. Wu Y, Lim J, Yang MH. Online object tracking: a benchmark. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE; 2013. p. 2411–8. doi:10.1109/CVPR.2013.312.
76. Kristan M, Matas J, Leonardis A, Vojíř T, Pflugfelder R, Fernandez G, et al. A novel performance evaluation methodology for single-target trackers. IEEE Trans Pattern Anal Mach Intell. 2016;38(11):2137–55. doi:10.1109/TPAMI.2016.2516982.
77. Holzinger A, Lukač N, Rozajac D, Johnston E, Kocic V, Hoerl B, et al. Enhancing trust in automated 3D point cloud data interpretation through explainable counterfactuals. Inf Fusion. 2025;119(4):103032. doi:10.1016/j.inffus.2025.103032.