IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

RefDiffMap: Diffusion-Guided Progressive Refinement for Vectorized HD Map Construction

Wenjie Gao[1], Entao Chang[1], Jiawei Fu[2], Ziyu Zhu[1], Shitao Chen[1], Nanning Zheng[1]

***Abstract*—High-definition (HD) map learning serves as an essential component of autonomous driving scene understanding, providing structured priors for planning and prediction. Recent transformer-based methods regress vectorized map elements via deformable attention over Bird's-Eye View (BEV) features. They typically employ a single-pass paradigm, starting from a set of initial queries. However, these queries struggle to precisely localize map elements within the large-scale BEV space. This difficulty is severely amplified when using lightweight backbones that produce less distinctive features. To address this, we propose RefDiffMap, which recasts map construction as a progressive refinement process driven by a diffusion model. We introduce a novel denoising query generator that, at each step, leverages the intermediate noisy geometry to sample relevant features from adaptive BEV RoIs. These features are distilled into context-aware queries that guide the decoder's next refinement. This creates a powerful geometry-feature co-evolution loop, allowing the model to iteratively correct localization errors. Comprehensive experiments show that RefDiffMap achieves competitive performance on the nuScenes and Argoverse 2 datasets. Notably, its robustness is highlighted with a ResNet-18 backbone, where it improves mAP by a significant 11.3% over our baseline MapTRv2. Further ablation studies validate the effectiveness of our approach.**

***Index Terms*—Autonomous Vehicle Navigation, Deep Learning for Visual Perception, Mapping**

I. INTRODUCTION

HIGH-DEFINITION (HD) maps are essential to autonomous driving, furnishing precise geometric, topological, and semantic abstractions of the environment. Because traditional manually curated, offline pipelines are costly and slow to update, recent work [1], [2], [3] pivots toward online map construction and represents map elements as sequences of vectorized points, a form naturally compatible with planning and prediction. Building on this paradigm, Transformer-based approaches employ point-based queries within deformable attention [4] to flexibly capture diverse geometries.

However, prevailing Transformer-based approaches rely on a single-pass decoding paradigm, wherein a set of learned queries directly regresses the full set of polyline coordinates in a single pass. This paradigm implicitly assumes that: (i) the initial query anchors are already positioned in close proximity to their geometric targets, and (ii) the underlying BEV features possess sufficient discriminative power for the sparse sampling of deformable attention to precisely localize thin or fragmented map elements. In practice, these assumptions are often violated, especially when using computationally efficient lightweight backbones (e.g., ResNet-18). Even minor initial misalignments in query anchors force the attention mechanism to search over a wide area. If the features within this expanded region lack saliency—a common issue for narrow lane dividers or pedestrian crossings—the attention mechanism is prone to attending to off-target regions.

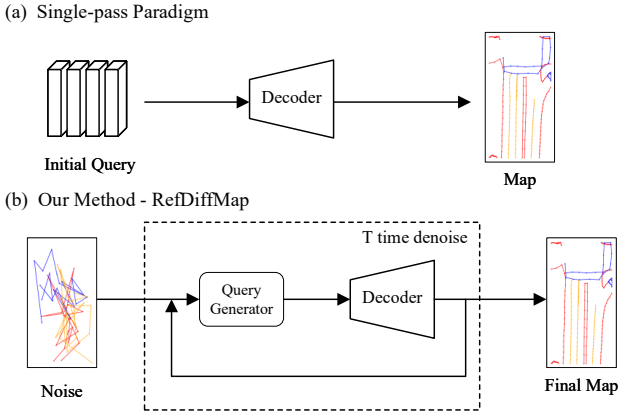

Fig. 1. (a) Single-pass methods regress the map from queries in a single pass, a process fragile to initial errors. (b) RefDiffMap implements a robust co-evolution loop where its query generator leverages intermediate geometry to guide feature resampling.

These accumulated errors highlight the need to move beyond single-pass decoding toward a multi-step refinement paradigm that supports iterative alignment between geometry and features. While recent efforts to incorporate diffusion-style denoising into HD mapping seem to provide such an iterative framework, they do not fully address the core challenge of geometry-feature alignment. For example, some approaches [5], [6] apply diffusion only at the raster level, merely smoothing BEV segmentation masks. Others [7], [8] use diffusion on coordinate sets for tasks like handling permutations or generating hypotheses, but they treat denoising as a latent point-set operation rather than a way to re-ground predictions. Critically, these methods do not leverage intermediate noisy geometry to actively re-query BEV features, thus missing the essential feedback loop for progressively refining geometric estimates and correcting localization errors.

Manuscript received: September 24, 2025; Accepted: December 1, 2025.

This paper was recommended for publication by Editor Javier Civera upon evaluation of the Associate Editor and Reviewers' comments.

This work was supported by the National Natural Science Foundation of China (Grant No. 62088102).

[1]W. Gao, E. Chang, Z. Zhu, S. Chen† (corresponding author), and N. Zheng are with National Key Laboratory of Human-Machine Hybrid Augmented Intelligence, National Engineering Research Center for Visual Information and Applications, and Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Shaanxi 710049, P.R. China. {gaowenjie999, qq969827455, zhuzy_2016, chenshitao, nnzheng}@mail.xjtu.edu.cn

[2]J. Fu is with The Chinese University of Hong Kong, Shatin, Hong Kong. jwfu@cse.cuhk.edu.hk

Digital Object Identifier (DOI): see top of this page.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Consequently, a key research gap persists: the absence of a multi-step alignment loop deeply integrated into the vector decoding process. To address this, we introduce RefDiffMap, a novel diffusion framework that transforms each denoising timestep into an explicit co-evolution of geometry and features. Specifically, we design a denoising query generator that creates dynamic, semantically-aware queries from intermediate noisy geometry. This is done by first defining local search regions around the current geometric estimates and then extracting salient BEV features from those regions. These queries condition a denoising head, guiding it to focus on the most informative features while expanding the search for weak or fragmented structures. This iterative "propose-and-resample" loop forms a true geometry-feature co-evolution, evolving the diffusion process from passive point-set refinement into an active mechanism for systematic error correction.

Experiments on nuScenes [9] and Argoverse 2 [10] show that RefDiffMap boosts online HD map reconstruction quality, with pronounced benefits when using lightweight backbones like ResNet-18. Comprehensive ablation studies validate our central claim: the proposed geometry-feature co-evolution loop is the key driver of these improvements. Our contributions are as follows.

- We propose RefDiffMap, a diffusion-based framework that repurposes the denoising process for iterative map geometric refinement.
- We design a denoising query generator to realize a geometry-feature co-evolution loop, where intermediate noisy geometry acts as explicit queries to guide feature resampling.
- Our method achieves highly competitive performance on nuScenes and demonstrates exceptional robustness and efficiency with lightweight backbones.

II. RELATED WORK

A. HD Map Construction

The task of online HD map construction was pioneered by HDMapNet [11], which aimed to replace laborious manual annotation[12], [13]. Subsequently, VectorMapNet [3] introduced an end-to-end solution that directly predicted vectorized representations, avoiding complex post-processing. This vectorized format is naturally compatible with downstream planning modules [14]. The foundational paradigm for many current methods was established by MapTR [1] and its successor MapTRv2 [2], which modeled map elements as queries and employed deformable attention to interact with BEV features serving as keys and values. Building on this, a series of works focused on refinement; for instance, BeMapNet [15], PivotNet [16], and GeMap [17] improved performance by optimizing the modeling of road element midpoints for better geometric fitting. Another line of work, including StreamMapNet [18] and MapTracker [19], achieved further breakthroughs by incorporating temporal information, leveraging elements from previous frames to guide generation in the current one. Additionally, methods [20], [21], [22] like NMP have explored using auxiliary information to aid detection. However, a common limitation of these methods is their reliance on a single-pass regression paradigm, which is susceptible to suboptimal initial queries that lead to inaccurate feature sampling and poor localization. Breaking from this paradigm, RefDiffMap introduces a diffusion-based framework for progressive, iterative refinement, leading to a significant performance breakthrough.

B. Diffusion Model for Object Detection

The application of diffusion models in detection was notably advanced by DiffusionDet [23], which reframed the task as a noise-to-box denoising process. This paradigm was extended to 3D space, leading to diverse strategies. For instance, Zhou et al. [24] employed diffusion models to generate 3D proposals from random Gaussian distributions. In contrast to generation from scratch, DiffuBox [25] proposed a diffusion-based refinement approach, leveraging LiDAR point clouds to jointly optimize coarse initial boxes and enhance localization accuracy. Beyond direct box generation and optimization, diffusion models have also been integrated into training pipelines. Diffusion-SS3D [26] and Diff3DETR [27] incorporate them into semi-supervised 3D detection frameworks to generate high-quality pseudo-labels. More recently, this trend has permeated HD map construction. DiffMap [5] uses diffusion to smooth and rectify BEV segmentation masks at the rasterized level. Others operate directly on vectorized coordinates; for instance, PolyDiffuse [7] refines geometry from coarse proposals, while MapDiffusion [8] generates diverse map hypotheses from noise to estimate uncertainty. However, the iterative process in these methods is largely treated as a latent operation on coordinate sets. They fail to leverage the intermediate geometries produced during diffusion to actively and repeatedly query BEV features. Consequently, they lack a crucial feedback loop that tightly couples geometric estimation with feature extraction.

C. Query Denoising

Query denoising has emerged as a powerful technique for stabilizing and accelerating the training of DETR-style models. DN-DETR [28] first introduced this concept by adding noise to ground-truth boxes during training, forcing the model to reconstruct the original boxes and thus simplifying the bipartite matching process. This idea was further advanced by DINO [29] and MaskDINO [30], which refined the denoising task for improved detection and segmentation. In a specialized application, DN-MOT [31] customized a denoising strategy to improve the accuracy and robustness of multiple object tracking in autonomous driving. In the context of HD map construction, SQD-MapNet [32] leverages query denoising as a training-time auxiliary task to learn temporal consistency. In contrast, our work repurposes denoising as a core inference mechanism within a diffusion framework, where our proposed denoising query generator enables a geometry-feature co-evolution loop for progressive refinement.

III. METHOD

A. Preliminary

HD Map Modeling. In vector-based representation of HD maps, the detection output for a scenario is denoted as $\mathcal{X} \coloneqq$

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

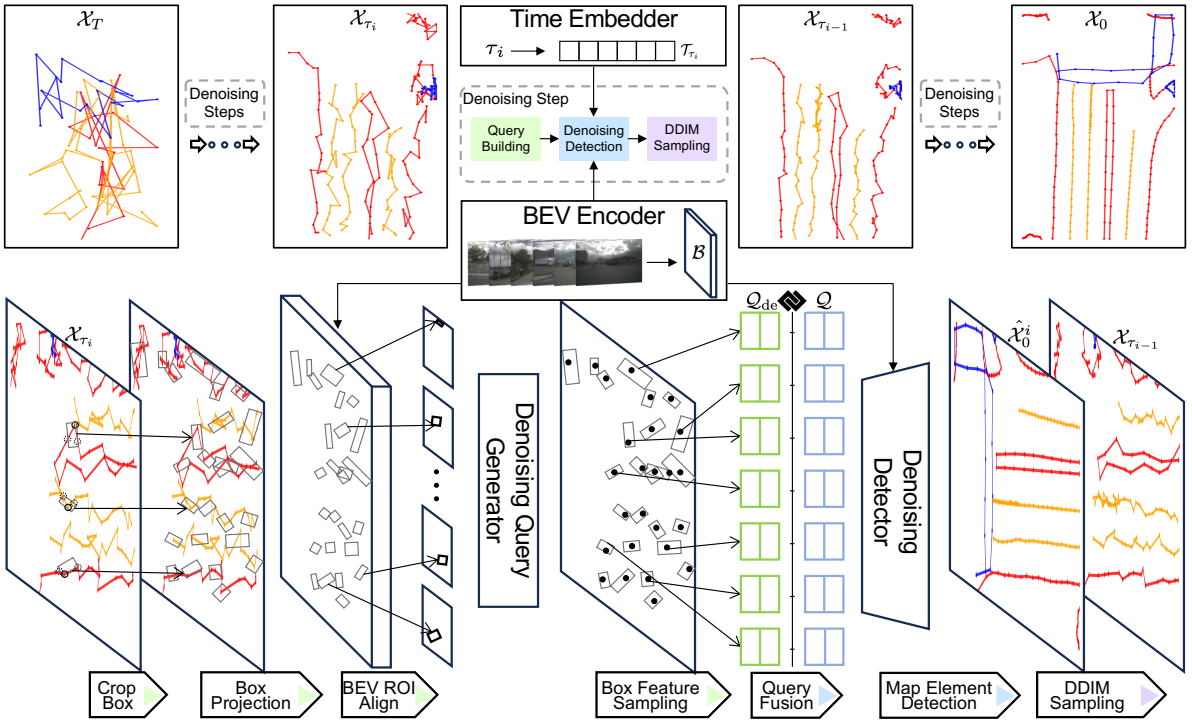

Fig. 2. Denoising steps of RefDiffMap with denoising query generation process. RefDiffMap applies multi-step DDIM sampling to get the final HD map elements $\mathcal{X}_0$ from pure Gaussian noise $\mathcal{X}_T$. In each denoising step, we first generate denoising queries from $\mathcal{X}_{\tau_i}$ using its cropped box in the BEV feature, then the combined query and position are processed by our denoising detector to predict $\hat{\mathcal{X}}_0^i$, based on which we propagate $\mathcal{X}_{\tau_{i-1}}$ through DDIM sampling.

$\{\mathbf{x}_n\}_{n=1}^N$, where each map element $\mathbf{x}_n := \{\{\mathbf{p}_m\}_{m=1}^M, c\}$ consists of M predefined points $\mathbf{p}_m = (x, y)$ and a category label c (e.g., pedestrian crossings, boundaries, or dividers). This vectorized format enables precise reconstruction of road elements from sensor data. Existing methods, inspired by DETR [4], employ hierarchical queries $\mathcal{Q} := \{\{q_n^m\}_{m=1}^M\}_{n=1}^N$ at instance and point levels to regress these elements. Instance-level queries capture overall object properties, while point-level queries refine coordinates.

Diffusion Model. Diffusion models are generative frameworks that simulate a bidirectional process: a forward diffusion phase gradually adds Gaussian noise to clean data, transforming it into pure noise, and a reverse denoising phase reconstructs the original data iteratively. Formally, denoising diffusion probabilistic models (DDPM) [33] define the forward process as a Markov chain that perturbs data $\Psi_0 \sim q(\Psi_0)$ over T timesteps:

$$q(\Psi_t|\Psi_{t-1}) = \mathcal{N}(\Psi_t; \sqrt{1-\beta_t}\Psi_{t-1}, \beta_t\mathbf{I}), \tag{1}$$

with a closed-form expression for any t:

$$q(\Psi_t|\Psi_0) = \mathcal{N}(\Psi_t; \sqrt{\bar{\alpha}_t}\Psi_0, (1-\bar{\alpha}_t)\mathbf{I}), \tag{2}$$

where $\bar{\alpha}_t = \prod_{s=1}^t(1-\beta_s)$ and β_t controls the noise schedule. Thus, Ψ_t can be sampled as $\Psi_t = \sqrt{\bar{\alpha}_t}\Psi_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$, with $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

The reverse process starts from noise $\Psi_T \approx \mathcal{N}(\mathbf{0}, \mathbf{I})$ and uses a learned model $\epsilon_\theta(\Psi_t, t, \mathbf{c})$ to predict the added noise, enabling iterative denoising:

$$\Psi_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left[\Psi_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\Psi_t, t, \mathbf{c})\right] + \sigma_t\mathbf{z}, \tag{3}$$

where $\alpha_t = 1-\beta_t$, $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and σ_t matches the forward variance (e.g., β_t). This paradigm has been successfully applied to object detection, where the model learns to denoise a set of bounding boxes (Ψ_0) conditioned on image features ($\mathbf{c}$). Analogously, in our work, we frame HD map generation as a conditional denoising process, where the target to be generated, Ψ_0, is the set of map vectors $\mathcal{X}$.

B. Model Architecture

Building upon the diffusion model paradigm, we introduce RefDiffMap, a novel framework designed to generate HD maps through a conditional denoising process. As illustrated in fig. 2, RefDiffMap restores final detection result $\mathcal{X}_0$ from pure noise $\mathcal{X}_T$ with DDIM sampling at selected denoising steps $\{t_{\tau_n}, t_{\tau_{n-1}}..., t_{\tau_0}\}$. Our framework contains four key modules: a BEV encoder, a time embedder, our proposed denoising query generator, and a denoising detector.

First, we generate the BEV feature $\mathcal{B}$ as the conditioning input using the BEV encoder. At each denoising step t_{τ_i}, the denoising query generator extracts the denoising query $\mathcal{Q}_{\text{de}}$ based on $\mathcal{X}_{\tau_i}$, which is then fused with the initial query $\mathcal{Q}$ to form the final query. Subsequently, the time-shifted BEV feature and the combined query are fed into the denoising detector to predict the clean result $\hat{\mathcal{X}}_0^i$ and compute $\mathcal{X}_{\tau_{i-1}}$ via DDIM sampling.

BEV Encoder The BEV encoder transforms multi-view images into a BEV feature, denoted as $\mathcal{B} \in \mathbb{R}^{H\times W\times C}$, which is further leveraged in the denoising detector and denoising query generation process. Following established practices, we employ a ResNet-50 [34] backbone to extract image features,

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

which are then projected into BEV space using the perspective-to-BEV transformation module from BEVFormer [35]. Additionally, we utilize a ResNet-18 backbone to validate the effectiveness of our method on a more lightweight architecture.

Time Embedder To make the network aware of the current noise level, we use a time embedder to encode the denoising timestep t_{τ_i} into a time embedding $\mathcal{T}_{\tau i}$. Similar to prior work [23], we first convert the discrete timestep into a continuous representation using sinusoidal position embeddings, which is then processed by a MLP. The resulting embedding is used to modulate the BEV feature $\mathcal{B}$ via adaptive scaling and shifting, producing the time-aware feature $\mathcal{B}\tau_i$.

Denoising Detector The denoising detector is the core prediction engine of our framework. Its function is to predict a clean HD map $\hat{\mathcal{X}}_0^i$ from the time-aware BEV feature $\mathcal{B}_{\tau_i}$ and the fused input queries. Architecturally, we build upon the Transformer-based detector from MapTRv2 [2], which utilizes a hierarchical query design and deformable attention to regress map elements. The detector takes the fused queries and $\mathcal{B}\tau_i$ as input, and after several decoder layers, outputs the predicted clean map elements through its classification and regression heads.

C. Denoising Query Generator

A key component for achieving iterative geometry-feature alignment in our framework is the Denoising Query Generator, which dynamically produces a set of context-aware queries, $\mathcal{Q}_{\text{de}}$, from the intermediate noisy map prediction $\mathcal{X}_{\tau_i}$. This module distills salient semantic and spatial information from the Bird's-Eye-View (BEV) feature map $\mathcal{B}$ to guide the subsequent denoising step, optimizing the alignment between $\mathcal{X}_{\tau_i}$ and the denoising query for better conservation of features across iterations.

The process begins by cropping a Region of Interest (RoI) box $\mathbf{b}$ for each point $\mathbf{p}$ within $\mathcal{X}_{\tau_i}$. The spatial extent of each RoI is determined by aggregating its three nearest neighboring points, capturing the immediate local context. These RoI boxes, $\{\mathbf{b}_k\}_{k=1}^{N\cdot M}$, are then projected onto the BEV feature map $\mathcal{B}$ via box projection. Using a BEV RoIAlign mechanism similar to RoIAlign, we extract a corresponding feature patch $\mathbf{B}_b \in \mathbb{R}^{H_b \times W_b \times C}$ for each RoI, encapsulating rich semantic features around the noisy point. To form a compact query, a lightweight MLP predicts a salient reference coordinate $\mathbf{i} = \{h, w\}$ within $\mathbf{B}_b$, identifying the location with the maximum feature activation. A fixed-dimensional feature vector is then extracted via grid sampling centered at this reference coordinate, serving as the content of the denoising query. The resulting query set $\mathcal{Q}_{\text{de}}$ is adapted to match the dimensionality and spatial positioning of the detector's learnable queries. We validate the effectiveness of this point box strategy over the alternative feature extraction method in our ablation studies (section IV-D).

By providing a semantically rich and spatially accurate prior derived from the previous noisy output, $\mathcal{Q}_{\text{de}}$ initializes reference points in the deformable attention mechanism from an informed position. This enhances the exploration range for tiny or densely clustered HD map elements, leading to more accurate refinement throughout the iterative denoising process and improved performance of the final $\mathcal{X}_0$.

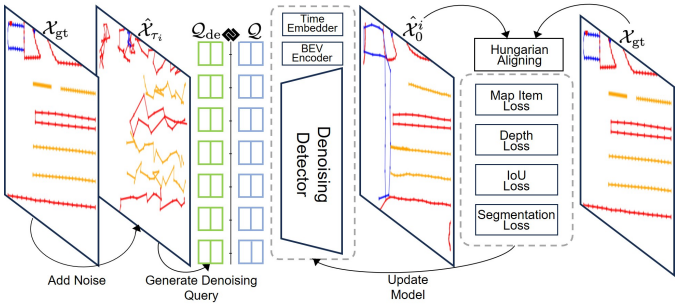

Fig. 3. Training pipeline of RefDiffMap. Given one case with the ground truth HD map elements $\mathcal{X}_{\text{gt}}$, we propagate the noise into it from the randomly selected timestep to get $\mathcal{X}_{\tau_j}$, which are then processed to generated denoising query and input to the denoising detector. The final loss will be calculated through $\hat{\mathcal{X}}_0^j$ and $\mathcal{X}_{\text{gt}}$ to update all the models after Hungarian aligning.

D. Denoising Inference

RefDiffMap models the HD map construction as a denoising process, which is decomposed as a sequence of discrete timesteps $\{t_{\tau_n}, t_{\tau_{n-1}}, ..., t_{\tau_0}\}$. Each sampling step is responsible for crop $\mathcal{X}_{\tau_{i-1}}$ from $\mathcal{X}_{\tau_i}$ conditioned on $\mathcal{B}$ and time embedding t_{τ_i}. As depicted in fig. 2, the refinement at each step involves three sequential stages: query building, denoising detection, and DDIM sampling.

Query Building. The first stage, query building, fuses the generated denoising query $\mathcal{Q}_{\text{de}}$ with the model's learnable query $\mathcal{Q}$ via element-wise multiplication. The denoising query $\mathcal{Q}_{\text{de}}$ is dynamically derived from the previous noisy estimate $\mathcal{X}_{\tau i}$, providing instance-specific context. In contrast, $\mathcal{Q}$ contains general geometric priors learned by the detector. By leveraging the noisy input $\mathcal{X}_{\tau i}$, this mechanism allows the detector to better distinguish between noise-induced anomalies and genuine map features.

Denoising Detection. In the second stage, the denoising detector predicts a clean version of the map, $\hat{\mathcal{X}}0^i$, from the noisy inputs. Specifically, the time embedding $\mathcal{T}\tau_i$ modulates the BEV feature $\mathcal{B}$ to produce a time-aware feature $\mathcal{B}\tau i$, informing the detector of the current noise level. The detector then takes the fused queries (both content and positional information from $\mathcal{Q}$de and $\mathcal{Q}$) and $\mathcal{B}\tau_i$ as input to output the clean map estimate $\hat{\mathcal{X}}_0^i$, following the architecture described in section III-B.

DDIM Sampling Finally, the detected results $\hat{\mathcal{X}}_0^i$ at t_{τ_i} from the detector is applied with $\mathcal{X}_{\tau_i}$ and τ_i together to sample $\mathcal{X}_{\tau_{i-1}}$ using DDIM:

$$\begin{aligned}\mathcal{X}_{\tau_{i-1}} =& \sqrt{\bar{\alpha}_{\tau_{i-1}}}\hat{\mathcal{X}}_0^i \\ &+ \sqrt{1-\bar{\alpha}_{\tau_{i-1}}-\sigma_{\tau_i}^2}\,\frac{\mathcal{X}_{\tau_i}-\sqrt{\bar{\alpha}_{\tau_i}}\hat{\mathcal{X}}_0^i}{\sqrt{1-\bar{\alpha}_{\tau_i}}} \\ &+ \sigma_{\tau_i}\mathbf{z},\end{aligned} \tag{4}$$

where $\bar{\alpha}_{\tau_{i-1}}$, $\sigma_{\tau_i}^2$ are the pre-defined hyperparameters in the sampling process of diffusion model, and $\mathbf{z} \in \mathcal{N}(\mathbf{0}, \mathbf{I})$. The sampling process will not end until we get $\mathcal{X}_0$, which serves as the final HD map construction results.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TABLE I
PERFORMANCE COMPARISON ON NUSCENES DATASET.

Methods	Venue	Modality	Backbone	Seg. Loss	$\text{AP}_{div}(\uparrow)$	$\text{AP}_{ped}(\uparrow)$	$\text{AP}_{bnd}(\uparrow)$	mAP(↑)
VectorMapNet[3]	ICML'23	C	R50		42.5	51.4	44.1	46.0
MapTR[1]	ICLR'23	C	R50		59.8	56.2	60.1	58.7
HDMapNe[11]	ICRA'22	C	EB0	✓	14.4	21.7	33.0	23.0
PivotNet[16]	ICCV'23	C	R50	✓	58.8	53.8	59.6	57.4
BeMapNet[15]	CVPR'23	C	R50	✓	62.3	57.7	59.4	59.8
StreamMapNet[18]	WACV'23	C	R50	✓	60.4	61.9	58.9	60.4
MapTRv2[2]	IJCV'24	C	R50	✓	59.8	62.4	62.4	61.5
PolyDiffuse[7]	NeurIPS'24	C	R50	✓	58.2	59.7	61.3	59.7
HDMapNet[11]	ICRA'22	C & L	EB0 & PP		29.6	16.3	46.7	31.0
VectorMapNet[3]	ICML'23	C & L	R50 & PP		60.1	48.2	53.0	53.7
MapTR[1]	ICLR'23	C & L	R50 & Sec		62.3	55.9	**69.3**	62.5
RefDiffMap (Ours)	-	C	R50	✓	**64.0**	**64.6**	64.3	**64.3**

TABLE II
PERFORMANCE COMPARISON ON ARGOVERSE 2 DATASET.

Methods	$\text{AP}_{div}(\uparrow)$	$\text{AP}_{ped}(\uparrow)$	$\text{AP}_{bnd}(\uparrow)$	mAP(↑)
VectorMapNet[3]	36.1	38.3	39.2	37.9
HDMapNet[11]	5.7	13.1	37.6	18.8
PivotNet[16]	31.3	47.5	43.4	40.7
StreamMapNet[18]	62.0	59.5	63.0	61.5
MapTRv2[2]	60.7	**68.9**	64.5	64.7
RefDiffMap (Ours)	**69.5**	61.7	**64.7**	**65.3**

E. Denoising Training

Training Pipeline RefDiffMap follows the standard training process from DiffusionDet [23], and the training pipeline of RefDiffMap is shown in fig. 3. Consistent with the sampling process, we get the BEV feature using the encoder as $\mathcal{B}$. Then given a random denoising timestep t_{τ_j}, we add noise to the ground truth and propagate its desired noisy HD map elements $\mathcal{X}_{\tau_j} \coloneqq \sqrt{\bar{\alpha}_{\tau_j}}\mathcal{X}_{\text{gt}} + \sqrt{1-\bar{\alpha}_{\tau_j}}\epsilon$. Then we generate the denoising query $\mathcal{Q}_{\text{de}}$ from $\mathcal{X}_{\tau_j}$ using the box-cropped BEV feature $\mathbf{B}_b$ and the corresponding reference position $\mathbf{i}$. The fusion of $\mathcal{Q}_{\text{de}}$ and $\mathcal{Q}$ is pushed to the denoising detector to organize the detected HD map items $\hat{\mathcal{X}}_0^j$. We then calculate the loss between $\mathcal{X}_{\text{gt}}$ and $\hat{\mathcal{X}}_0^j$ with Hungarian algorithm. Different from previous methods [2] which only leverage the loss to update the detection or segmentation model in HD map learning, we backpropagate the overall loss to update the denoising detector, time embedder, denoising query generator, and BEV encoder together. Our target is to train all the modules to match and restore the ground truth HD map elements under the current added noise based on the randomly provided denoising timestep.

Loss Function We follow MapTRv2 [2] and calculate the loss function between $\hat{\mathcal{X}}_0^j$ and $\mathcal{X}_{\text{gt}}$ using:

$$\mathcal{L} \coloneqq \lambda_1\mathcal{L}_{\text{map}} + \lambda_2\mathcal{L}_{\text{IoU}} + \lambda_3\mathcal{L}_{\text{depth}} + \lambda_4\mathcal{L}_{\text{seg}}, \tag{5}$$

where $\mathcal{L}_{\text{map}}$ is combined through point-to-point loss, box loss, and direction loss, $\mathcal{L}_{\text{IoU}}$ means IoU loss based on detection results, $\mathcal{L}_{\text{depth}}$ and $\mathcal{L}_{\text{seg}}$ are the auxiliary losses means the depth estimation loss from the images and the segmentation loss from BEV, respectively. λ means the weights of losses.

IV. EXPERIMENT

A. Experimental Setup

Datasets We evaluate RefDiffMap primarily on nuScenes [9] following prior methods [1]. It includes 1000 scenes of 20s duration, with 2D vectorized maps and RGB images from 6 cameras covering 360 degree FOV. We also test on Argoverse 2 [10], with 1000 logs of 15s 20Hz RGB images from 7 cameras and 3D vectorized maps.

Metrics The evaluation area covers $[-15.0m, 15.0m]$ along the X-axis and $[-30.0m, 30.0m]$ along the Y-axis. We use Average Precision (AP) to measure map construction quality, with Chamfer distance $\text{D}_{chamfer}$ as the matching criterion between predictions and ground truth. The mean Average Precision (mAP) is computed across multiple thresholds $\xi \in \{0.5, 1.0, 1.5\}$. Following standard practice, we evaluate three map element types: pedestrian crossings, lane dividers, and road boundaries.

Implementation Details The total diffusion timesteps are set to 100, where we randomly select one timestep to train the model for each training case. In the training process, all the models are trained with batch size 24 on 2-3 GPUs. The inference speed was evaluated on NVIDIA RTX 4090 GPU and Intel Xeon Gold 6326 CPU. The default training schedule is 117K iterations on nuScenes and 136K iterations on Argoverse 2. We adopt the AdamW optimizer with a learning rate of 6×10^{-4}. At the denoising sampling stage, we report performances of RefDiffMap under diverse sampling steps under the same hyperparameters schedule as training. Since we adopt MapTRv2 [2] as our baseline detector, we primarily focus on comparisons with MapTRv2.

B. Main Results

Performance on nuScenes Dataset. We compare RefDiffMap with prior vision-based methods on the nuScenes validation set. As shown in table I, with 3 denoising steps, our method achieves a mAP of 64.3 using a camera-only setup and

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TABLE III
PERFORMANCE COMPARISON WITH LIGHTWEIGHT BACKBONE ON NUSCENES DATASET.

Methods	Backbone	Sampling Steps	$AP_{div}(\uparrow)$	$AP_{ped}(\uparrow)$	$AP_{bnd}(\uparrow)$	mAP(↑)
MapTR	R18	-	39.6	49.9	48.2	45.9
MapTRv2	R18	-	46.9	55.1	54.9	52.3
RefDiffMap	R18	1	48.6	54.8	55.1	52.9
RefDiffMap	R18	3	**57.2**	**58.8**	**58.5**	**58.2**
RefDiffMap	R18	5	55.9	57.9	58.2	57.3
RefDiffMap	R18	20	48.6	54.8	55.1	52.9

TABLE IV
ABLATION ON THE NUMBER OF DENOISING SAMPLING STEPS.

Sampling Steps	nuScenes				FPS	Argoverse 2				FPS
	$AP_{div}(\uparrow)$	$AP_{ped}(\uparrow)$	$AP_{bnd}(\uparrow)$	mAP(↑)		$AP_{div}(\uparrow)$	$AP_{ped}(\uparrow)$	$AP_{bnd}(\uparrow)$	mAP(↑)	
1	57.9	58.2	63.9	60.0	8.8	65.8	58.4	62.8	62.3	8.4
2	63.8	63.9	64.9	64.2	8.6	68.4	61.1	64.4	64.6	8.1
3	64.0	64.6	64.3	64.3	8.2	69.5	61.7	64.7	65.3	7.6
4	63.5	64.7	63.5	63.9	7.0	69.1	61.5	64.6	65.1	6.8
5	63.4	64.8	63.1	63.8	5.9	69.5	61.5	64.7	65.2	5.7
10	62.8	64.5	61.8	63.0	3.1	67.9	61.3	63.8	64.3	3.1
20	62.6	64.5	61.4	62.8	1.9	67.1	61.1	63.6	63.9	1.9
100	62.3	64.3	60.9	62.5	0.4	66.8	61.0	63.4	63.7	0.4

TABLE V
ABLATION ON QUERY EMBEDDING METHODS ON NUSCENES DATASET.

Methods	Steps	AP_{div}	AP_{ped}	AP_{bnd}	mAP
Grid Sampling	1	6.1	6.0	4.8	5.6
	3	51.1	55.8	59.5	55.5
	5	55.6	59.2	62.0	58.9
	20	61.2	62.1	64.4	62.5
Object Box	1	3.2	10.1	7.7	7.0
	3	41.0	61.3	52.9	51.7
	5	44.5	61.9	54.7	53.7
	20	47.9	62.0	56.3	55.7
Point Box	3	64.0	64.6	64.3	64.3

ResNet-50 backbone, outperforming existing approaches by a significant margin. It notably improves AP for lane dividers (+1.7 points) and pedestrian crossings (+2.2 points) over MapTRv2, and even surpasses some multi-modal methods fusing camera and LiDAR data. This gain stems from our denoising query generator, which enables precise localization via iterative refinement. Results for varying denoising steps are detailed in the ablation study.

Performance on Argoverse 2. On Argoverse 2, as shown in table II, RefDiffMap achieves a mAP of 65.3 with 3 denoising steps, outperforming MapTRv2 by 0.6 points. This validates our method's efficacy and flexibility across datasets in autonomous driving.

C. Effectiveness with Lightweight Backbones

To evaluate RefDiffMap's effectiveness with lightweight backbones, we conduct experiments using ResNet-18 on the nuScenes dataset. As shown in table III, our method consistently outperforms baselines across varying denoising steps. For instance, with 3 steps, it achieves a mAP of 58.2, improving by 5.9 points over MapTRv2 (52.3 mAP) and by 12.3 points over MapTR (45.9 mAP). Performance peaks at 3 steps and declines slightly with more (e.g., 57.3 mAP at 5 steps) or fewer (52.9 mAP at 1 step). Notably, the 5.9-point gain with ResNet-18 exceeds the 2.8-point improvement seen with ResNet-50, demonstrating how our diffusion-based approach compensates for less distinctive features in lightweight architectures through multi-step denoising.

D. Ablation Study

Ablation studies on nuScenes and Argoverse 2 datasets evaluate the contributions of our method's components. We focus on the effect of DDIM sampling steps and different denoising query generation methods.

Effect of Sampling Steps. DDIM enables faster sampling by adjusting step frequency, balancing gradual denoising with performance. As shown in table IV, increasing steps initially boosts accuracy: on nuScenes, 3 steps yield 64.3 mAP (+4.3 over 1 step); on Argoverse 2, +3.0 mAP. This demonstrates how multi-step denoising refines queries and corrects tiny map elements, outperforming single-pass baselines (e.g., 1 step degenerates to prior methods). However, beyond 5 steps, gains diminish (e.g., 63.0 mAP at 10 steps on nuScenes), likely due to overfitting to noise. Inference speed (FPS) decreases with more steps (e.g., from 8.8 FPS at 1 step to 0.4 FPS at 100 steps on nuScenes), highlighting a trade-off: 3 steps offer optimal accuracy-speed balance for complex scenes like nuScenes.

Effect of Denoising Query Embedding. We ablate query generation methods, comparing our point-box approach with grid sampling and object-box variants. Details follow.

Grid Sampling: This directly samples BEV features using intermediate noisy points. From $\mathcal{X}_{\tau_i}$, we extract points $\{\{\mathbf{p}_n^m\}_{m=1}^M\}_{n=1}^N$ and grid-sample on $\mathcal{B}$ to form $\mathcal{Q}_{\text{de}}$, which fuses with initial query $\mathcal{Q}$ for denoising.

Object Box: Inspired by SAM-DETR [36], this computes instance-level bounding boxes $\mathbf{b} = \{x_{\min}, x_{\max}, y_{\min}, y_{\max}\}$ from all points in each element $\mathbf{x}$. RoIs $\{\mathbf{b}_n\}_{n=1}^N$ are projected onto $\mathcal{B}$, yielding patches $\mathbf{B}_b \in \mathbb{R}^{H_b \times W_b \times C}$. An MLP predicts

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

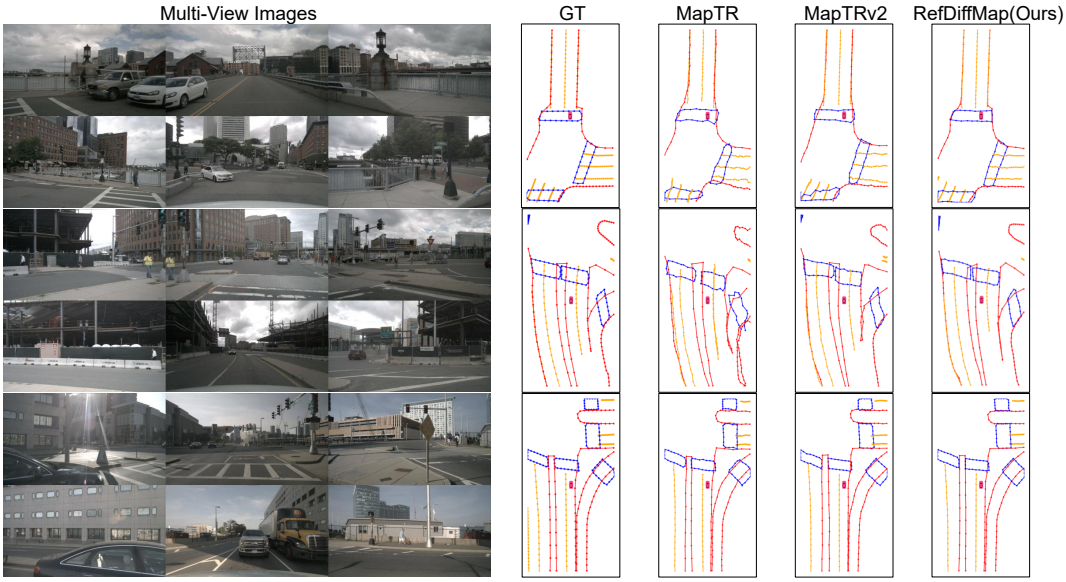

Fig. 4. Comparison with the HDMapNet, Maptrv2, and ground truth on qualitative visualization under different scenarios. In the HD-map, we show *road boundaries*, *lane dividers*, and *pedestrian crossings*.

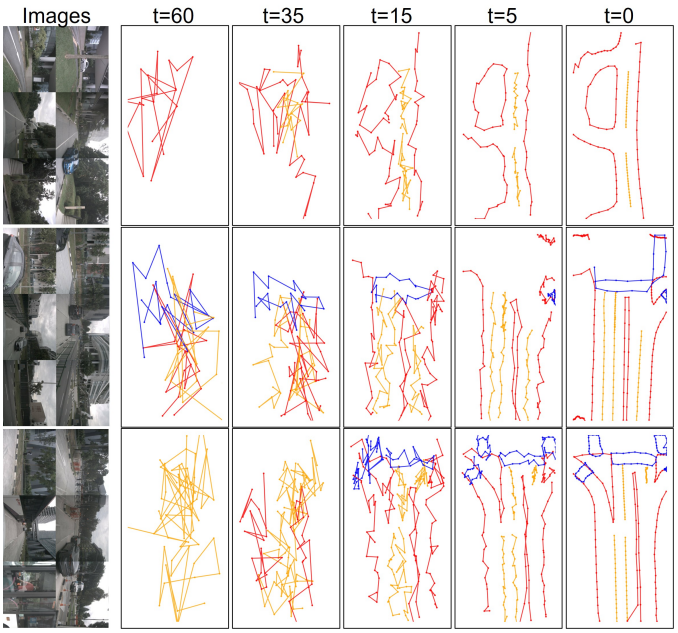

Fig. 5. Visualization results of in the timesteps 60, 35, 15, 5, 0 in the denoising process for several driving scenarios.

reference positions $\mathbf{i} \in \mathbb{R}^{M\times 2}$, enabling grid sampling to generate per-point queries, assembled into $\mathcal{Q}_{\text{de}}$.

Point Box (Ours): As detailed in the main text, this crops RoIs per point using its three nearest neighbors, extracting salient features for compact, context-aware queries.

As shown in table V, our point-box method achieves the best performance (64.3 mAP at 3 steps), surpassing grid sampling (+8.8 points) and object box (+12.6 points). Its moderate search range suits sparse HD map distributions, enabling better geometry-feature alignment than direct sampling (grid) or sparse instance-level boxes (object).

E. Qualitative Analysis

The qualitative analysis contains two parts, the main visualization results with a comparison of other methods and the visualization results in the DDIM sampling.

Visualization Results in Denoising Process We also provide some intermediate results of the denoising process in fig. 5. Our HD map learning process is a denoising process from pure noise to final results with the corresponding timestep from the total denoising step T to 0. We select three complex cases of the nuScenes dataset and select intermediate timesteps 60, 35, 15, and 5 with the final results for visualization. The visualization results show that during the denoising process, the HD map components are becoming more clear and complete gradually. The final results at $t = 0$ have gotten rid of all the noise.

Main Visualization Results We show some qualitative comparisons of RefDiffMap with ground truth, MapTR [1] and MapTRv2 [2] in nuScenes datasets in fig. 4. We can observe that RefDiffMap can restore HD map details more vividly and clearly, especially those with complicated structures and objects. Compared with other methods, RefDiffMap utilizes the diffusion model to calculate more precise tiny HD map components and propose better performance.

V. CONCLUSION AND DISCUSSION

In this paper, we introduce RefDiffMap, a framework that reimagines HD map construction as an iterative refinement process, rather than a single-pass decoding task. Central to our method is a new denoising query generation technique, which drives a co-evolution loop between geometry and features: predictions from each step help resample more relevant image details for the next, steadily ironing out localization errors. Our experiments reveal that this iterative strategy delivers marked gains in map element accuracy, shining especially on tricky, fragmented structures and when combined with lightweight backbones that yield less distinctive features. Though it does

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

add some computational burden, we've shown this exchange is well worth it for the enhanced geometric precision. Ultimately, we think this points to fertile ground for future work on building stronger, more efficient iterative decoders in HD map learning.

REFERENCES

[1] B. Liao, S. Chen, X. Wang, T. Cheng, Q. Zhang, W. Liu, and C. Huang, "Maptr: Structured modeling and learning for online vectorized hd map construction," *arXiv preprint arXiv:2208.14437*, 2022.

[2] B. Liao, S. Chen, Y. Zhang, B. Jiang, Q. Zhang, W. Liu, C. Huang, and X. Wang, "Maptrv2: An end-to-end framework for online vectorized hd map construction," *International Journal of Computer Vision*, vol. 133, no. 3, pp. 1352–1374, 2025.

[3] Y. Liu, T. Yuan, Y. Wang, Y. Wang, and H. Zhao, "Vectormapnet: End-to-end vectorized hd map learning," in *International Conference on Machine Learning*. PMLR, 2023, pp. 22 352–22 369.

[4] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2020.

[5] P. Jia, T. Wen, Z. Luo, M. Yang, K. Jiang, Z. Liu, X. Tang, Z. Lei, L. Cui, B. Zhang, *et al.*, "Diffmap: Enhancing map segmentation with map prior using diffusion model," *IEEE Robotics and Automation Letters*, vol. 9, no. 11, pp. 9836–9843, 2024.

[6] X. Hong, S. Li, K. Zeng, H. Shi, B. Peng, K. Yang, and Z. Li, "Ts-cgnet: Temporal-spatial fusion meets centerline-guided diffusion for bev mapping," *arXiv preprint arXiv:2503.02578*, 2025.

[7] J. Chen, R. Deng, and Y. Furukawa, "Polydiffuse: Polygonal shape reconstruction via guided set diffusion models," *Advances in Neural Information Processing Systems*, vol. 36, pp. 1863–1888, 2023.

[8] T. Monninger, Z. Zhang, Z. Mo, M. Z. Anwar, S. Staab, and S. Ding, "Mapdiffusion: Generative diffusion for vectorized online hd map construction and uncertainty estimation in autonomous driving," *arXiv preprint arXiv:2507.21423*, 2025.

[9] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631.

[10] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. K. Pontes, *et al.*, "Argoverse 2: Next generation datasets for self-driving perception and forecasting," *arXiv preprint arXiv:2301.00493*, 2023.

[11] Q. Li, Y. Wang, Y. Wang, and H. Zhao, "Hdmapnet: An online hd map construction and evaluation framework," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 4628–4634.

[12] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[13] J. Zhang, S. Singh, *et al.*, "Loam: Lidar odometry and mapping in real-time." in *Robotics: Science and systems*, vol. 2, no. 9. Berkeley, CA, 2014, pp. 1–9.

[14] J. Shi, T. Zhang, S. Chen, N. Zheng, and J. Xin, "Modeling human-like driving behavior based on maximum entropy deep inverse reinforcement learning," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025, pp. 17 420–17 427.

[15] L. Qiao, W. Ding, X. Qiu, and C. Zhang, "End-to-end vectorized hd-map construction with piecewise bezier curve," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 13 218–13 228.

[16] W. Ding, L. Qiao, X. Qiu, and C. Zhang, "Pivotnet: Vectorized pivot learning for end-to-end hd map construction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3672–3682.

[17] Z. Zhang, Y. Zhang, X. Ding, F. Jin, and X. Yue, "Online vectorized hd map construction using geometry," in *European Conference on Computer Vision*. Springer, 2024, pp. 73–90.

[18] T. Yuan, Y. Liu, Y. Wang, Y. Wang, and H. Zhao, "Streammapnet: Streaming mapping network for vectorized online hd map construction," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 7356–7365.

[19] J. Chen, Y. Wu, J. Tan, H. Ma, and Y. Furukawa, "Maptracker: Tracking with strided memory fusion for consistent vector hd mapping," in *European Conference on Computer Vision*. Springer, 2024, pp. 90–107.

[20] X. Xiong, Y. Liu, T. Yuan, Y. Wang, Y. Wang, and H. Zhao, "Neural map prior for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 535–17 544.

[21] W. Gao, J. Fu, Y. Shen, H. Jing, S. Chen, and N. Zheng, "Complementing onboard sensors with satellite maps: a new perspective for hd map construction," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 11 103–11 109.

[22] R. Sun, L. Yang, D. Lingrand, and F. Precioso, "Mind the map! accounting for existing map information when estimating online hdmaps from sensor," *arXiv preprint arXiv:2311.10517*, 2023.

[23] S. Chen, P. Sun, Y. Song, and P. Luo, "Diffusiondet: Diffusion model for object detection," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 19 830–19 843.

[24] X. Zhou, J. Hou, T. Yao, D. Liang, Z. Liu, Z. Zou, X. Ye, J. Cheng, and X. Bai, "Diffusion-based 3d object detection with random boxes," in *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*. Springer, 2023, pp. 28–40.

[25] X. Chen, Z. Liu, K. Luo, S. Datta, A. Polavaram, Y. Wang, Y. You, B. Li, M. Pavone, W.-L. H. Chao, *et al.*, "Diffubox: Refining 3d object detection with point diffusion," *Advances in Neural Information Processing Systems*, vol. 37, pp. 103 681–103 705, 2024.

[26] C.-J. Ho, C.-H. Tai, Y.-Y. Lin, M.-H. Yang, and Y.-H. Tsai, "Diffusion-ss3d: Diffusion model for semi-supervised 3d object detection," *Advances in Neural Information Processing Systems*, vol. 36, pp. 49 100–49 112, 2023.

[27] J. Deng, J. Lu, and T. Zhang, "Diff3detr: Agent-based diffusion model for semi-supervised 3d object detection," in *European Conference on Computer Vision*. Springer, 2024, pp. 57–73.

[28] F. Li, H. Zhang, S. Liu, J. Guo, L. M. Ni, and L. Zhang, "Dn-detr: Accelerate detr training by introducing query denoising," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 13 619–13 627.

[29] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum, "Dino: Detr with improved denoising anchor boxes for end-to-end object detection," *arXiv preprint arXiv:2203.03605*, 2022.

[30] F. Li, H. Zhang, H. Xu, S. Liu, L. Zhang, L. M. Ni, and H.-Y. Shum, "Mask dino: Towards a unified transformer-based framework for object detection and segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 3041–3050.

[31] T. Fu, X. Wang, H. Yu, K. Niu, B. Li, and X. Xue, "Denoising-mot: Towards multiple object tracking with severe occlusions," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 2734–2743.

[32] S. Wang, F. Jia, W. Mao, Y. Liu, Y. Zhao, Z. Chen, T. Wang, C. Zhang, X. Zhang, and F. Zhao, "Stream query denoising for vectorized hd-map construction," in *European Conference on Computer Vision*. Springer, 2024, pp. 203–220.

[33] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[35] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Q. Yu, and J. Dai, "Bevformer: learning bird's-eye-view representation from lidar-camera via spatiotemporal transformers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[36] G. Zhang, Z. Luo, Y. Yu, K. Cui, and S. Lu, "Accelerating detr convergence via semantic-aligned matching," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 949–958.