Compact HD Map Construction via Douglas-Peucker Point Transformer

Ruixin Liu, Zejian Yuan*

Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, China
sweetylrx@stu.xjtu.edu.cn, yuan.ze.jian@xjtu.edu.cn

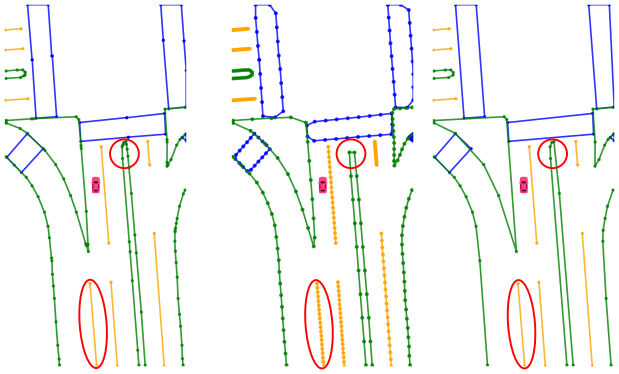
Abstract

High-definition (HD) map construction requires a comprehensive understanding of traffic environments, encompassing centimeter-level localization and rich semantic information. Previous works face challenges in redundant point representation or high-complexity curve modeling. In this paper, we present a flexible yet effective map element detector that synthesizes hierarchical information with a compact Douglas-Peucker (DP) point representation in a transformer architecture for robust and reliable predictions. Specifically, our proposed representation approximates class-agnostic map elements with DP points, which are sparsely located in crucial positions of structures and can get rid of redundancy and complexity. Besides, we design a position constraint with uncertainty to avoid potential ambiguities. Moreover, pairwise-point shape matching constraints are proposed to balance local structural information of different scales. Experiments on the public nuScenes dataset demonstrate that our method overwhelms current SOTAs. Extensive ablation studies validate each component of our methods. Codes will be released at https://github.com/sweety121/DPFormer.

Introduction

High-definition (HD) maps, as indispensable infrastructures for autonomous driving, provide centimeter-level traffic surroundings, diverse geometric shapes, and rich semantic information. Early works construct offline HD maps using SLAM-based methods (Shan and Englot 2018; Shan et al. 2020; Pan et al. 2021), incurring prohibitive maintenance costs. Due to the limited scalability, solutions for online HD map construction have witnessed remarkable development.

Most existing approaches either treat HD map construction as semantic segmentation tasks (Li et al. 2022a; Peng et al. 2023), disregarding geometry properties and instance associations of map elements, or focus on the sub-tasks such as lane detection (Liu et al. 2021a,b; Tabelini et al. 2021; Liu et al. 2022; Guan et al. 2023) and free space detection (Cao et al. 2021), which rely on specific views and are restricted by shape priors. Recently, several works (Liu et al. 2023b; Liao et al. 2023; Qiao et al. 2023) have been dedicated to producing vectorized HD maps capable of detecting various elements. Inspired by bird's eye view (BEV) perception methods (Philion and Fidler 2020; Li et al. 2022b; Chen et al. 2022b), BEV feature extractors and map element detectors are involved in typical pipelines.



(a) Original annotations (b) Equidistant points (c) DP points

Figure 1: Illustration of different element representations.

Despite notable advancements, current publicly available approaches still encounter significant challenges.

1) Compact representation for map element annotations.

MapTR (Liao et al. 2023) samples equidistant points along elements, which are redundant for simple shapes (marked by the red ellipse in Figure 1 (b)) but lack detailed depiction in critical parts with high curvature and sharp angles (marked by the red circle in Figure 1 (b)). BeMapNet (Qiao et al. 2023) represents map elements as piecewise Bézier curves, emphasizing the local optimal approximation while neglecting the global structures.

2) Point regression ambiguity for BEV perception.

Transforming features from image to BEV space is a routine procedure for HD map vectorization. Existing methods (Mallot et al. 1991; Li et al. 2022b; Peng et al. 2023; Philion and Fidler 2020) rely on either prior hypothesis or learnable parameters, restricted by problems such as distortion and deviation. Ambiguities in annotations are also amplified in early learning, hindering the final performance.

Motivated by above considerations, we propose a compact representation, utilizing the Douglas-Peucker (Douglas and Peucker 1973) algorithm to approximate map elements with as few points as possible, i.e., **D**ouglas-**P**eucker (DP) points. Contrary to existing redundant or piecewise repre-

sentations, DP points adopt a coarse-to-fine strategy, prioritizing the preservation of the overall structures without compromising accuracy (marked by the red ellipse and circle in Figure 1 (c)). Based on this representation, we design a transformer-based architecture, DPFormer, to detect map elements consisting of DP points end-to-end. Besides, given the potential ambiguity arising from the viewpoint transformation and DP annotation generation, we propose to provide an additional uncertainty estimation for position regression. Moreover, we introduce pairwise-point shape matching constraints to balance multi-scale positional and structural information. The main contributions of our work are threefold:

- We propose a class-agnostic representation to approximate map elements with DP points, which can be obtained by an elaborately designed DP point generation algorithm.
- An end-to-end framework is devised to fully aggregate both global and local information for reliable and robust map element detection.
- A position constraint with uncertainty estimation and pairwise-point shape matching constraints are proposed for better hierarchical supervision of map elements.

Related Work

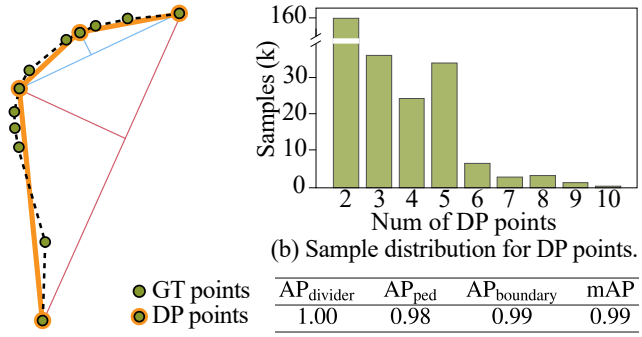
Lane detection. Lane detection, as a sub-task of HD map construction, has experienced outstanding progress, which provides reliable solutions in both parameterized and non-parameterized methods. Parameterized methods place particular emphasis on holistic and consistent structures of lane lines. LSTR (Liu et al. 2021b) models lane lines as a polynomial regression problem, and achieves swift inference taking inspiration from the end-to-end transformer architecture. BézierLaneNet (Feng et al. 2022) adopts actual control points of Bézier curves to improve the failure to the difficult-to-optimize and abstract polynomial coefficients.

Non-parameterized methods prioritize improving accuracy through flexible predictions. DSANet (Liu et al. 2023a) detects lane segments on uniformly divided grids, which achieves a performance boost through the incorporation of coupled shape matching constraints. Persformer (Chen et al. 2022a) unifies 2D and 3D lane detection in the BEV perception space, optimizing offsets with reference to predefined anchors. These single-view-based methods either rely on strong shape priors or require post-processing. By contrast, DPFormer synthesizes multi-view features into the BEV space and detects different map elements with arbitrary shapes in an end-to-end framework.

Vectorized HD map construction. In the context of BEV perception learning, vectorized HD map construction has become a research hotspot. HDMapNet (Li et al. 2022a) and BEVSegFormer (Peng et al. 2023) produce pixel-level segmentation maps, requiring sophisticated vectorization procedures for final results.

VectorMapNet (Liu et al. 2023b) first detects key points and then generates fine points in an auto-regressive way, serving as the first end-to-end method, but limited by long inference time and permutation ambiguity. MapTR (Liao et al. 2023) proposes a permutation-equivalent modeling method to eliminate ambiguity, and designs hierarchical queries to promote detection. Both of the two methods utilize equidistant point sets that might skip unanticipated visual details provided by complex structures. Moreover, they only constrain points for flexible predictions, regardless of multi-scale structures of pairwise points.

BeMapNet (Qiao et al. 2023) adopts piecewise Bézier curves to parameterize elements and designs masked cross-attention modules for exact queries. It divides each curve into segments by searching a local optimal approximation, which lacks sufficient consideration of overall contours. PivotNet (Ding et al. 2023) introduces pivot and collinear points to represent map element shapes. It models dynamic matching on pivot points, with fixed endpoints and point permutations. Despite competitive performances, the reconstructed ground truths decompose complex closed-shape elements into simpler ones, evading the challenge of tackling elements with diverse topologies. Unlike previous methods, DPFormer adopts DP points to achieve a coarse-to-fine approximation, uniformly representing both open-shape and closed-shape map elements.



(a) DP point representation. (b) Sample distribution for DP points.

$AP_{divider}$	AP_{ped}	$AP_{boundary}$	mAP
1.00	0.98	0.99	0.99

(c) Eval. on GTs of Max. 8 DP points.

Figure 2: Statistics on nuScenes for DP point representation.

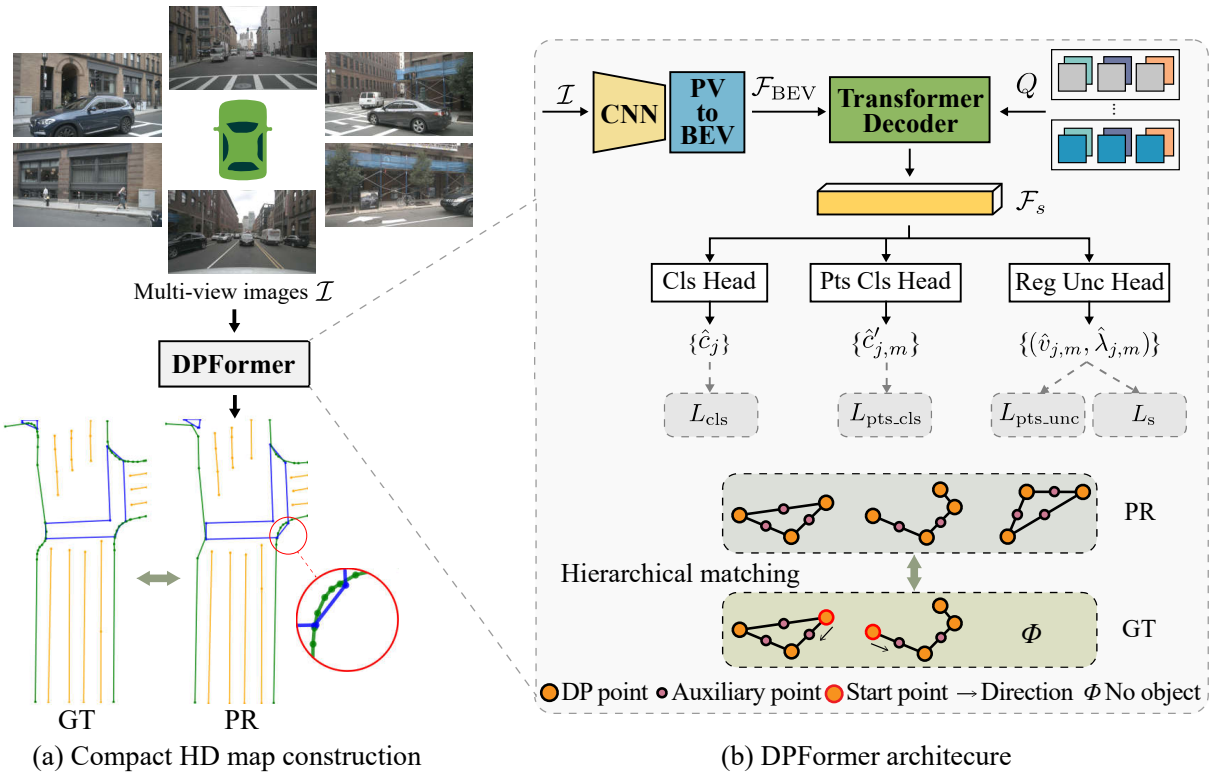
Methodology

DPFormer models compact HD maps in a unified DETR-like paradigm. Given that map elements vary in topologies and semantic information, discrete points have superiority in flexible depictions, including but not limited to open-shape polylines and closed-shape polygons.

Class-agnostic Element Representation with Douglas-Peucker Points

Feasibility analysis. Compact and concise point annotations play a vital role in efficient HD map construction. Neither piecewise Bézier curves (Qiao et al. 2023) nor dense equidistant points (Liao et al. 2023) sticks to this principle. We expect to provide a class-agnostic representation to depict map elements using a minimal number of points while preserving their overall shapes and detailed positions.

Douglas-Peucker algorithm (Douglas and Peucker 1973) is a common workaround to compress points. Specifically, we select the most representative DP points, and eliminate redundant points to meet approximation accuracy demands,



(a) Compact HD map construction (b) DPFormer architecure

Figure 3: Overall network architecture of DPFormer for compact HD map construction.

as indicated by orange in Figure 2 (a). According to statistics on the public nuScenes (Caesar et al. 2020) dataset, the overwhelming majority of elements can be accurately represented using fewer than 8 DP points within a tolerance of 1.5 m per element, as shown in Figure 2 (b). Moreover, we evaluate the widely recognized Chamfer distance (CD) on the DP points obtained from original annotations, achieving an astonishingly high mean average precision (mAP) exceeding 0.98, as shown in Figure 2 (c). *Results convincingly substantiate the feasibility of the representation with DP points.*

Douglas-Peucker points. The set of DP points is represented as $V^{\text{DP}} = \{v_0, \cdots, v_{n_d-1} | n_d \leq N\}$ (marked in orange in Figure 2 (a)), where n_d is the number of DP points, and N is its maximum value, preset according to dataset statistics. These DP points, varying in number, explicitly exist on map elements, distributed densely on the curved parts, and sparsely on the straight parts.

Auxiliary points. To better capture the visual cues around elements, we sample an additional set of auxiliary points $V^{\text{Aux}} = \{v_0, \cdots, v_{n_a-1} | n_a = M - n_d\}$, where M is fixed to indicate the number of all points that describe a single element, and n_a is the number of auxiliary points. The auxiliary points can be customized w.r.t. sharpness and curvature, or projected onto original annotations for utmost accuracy.

Both DP points V^{DP} and auxiliary points V^{Aux} form the ordered point set V for each map element. V is defined by $V = V^{\text{DP}} \cup V^{\text{Aux}}$ and $|V| = M$. With regard to fixed permutations' negative impact in training, we retain the equivalent permutation proposed by MapTR (Liao et al. 2023) for our representation, denoted as $\mathcal{V} = (V, \Gamma)$, where Γ is a group of equivalent permutations of V.

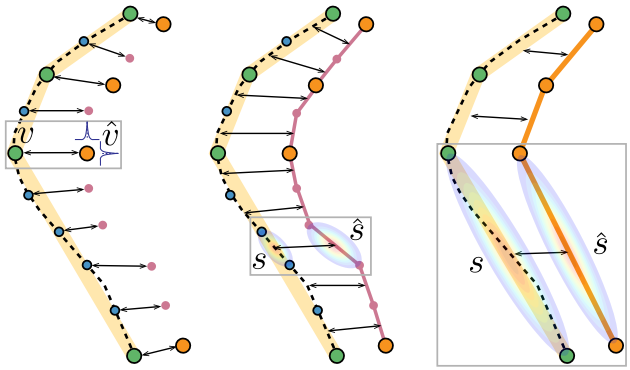
Network Architecture

The process of the compact HD map construction is illustrated in Figure 3(a). Our DPFormer architecture, comprising a BEV feature extractor and a map element detector, is shown in Figure 3(b) in detail.

BEV feature extractor. Given a set of K multi-view images $\mathcal{I} \in \mathbb{R}^{K \times H \times W \times 3}$, a shared CNN backbone is adopted to extract multi-view features $\mathcal{F} \in \mathbb{R}^{K \times H \times W \times C_1}$. We opt GKT (Chen et al. 2022b) to accomplish the transformation from perspective-view features $\mathcal{F}$ to BEV features $\mathcal{F}_{\text{BEV}} \in \mathbb{R}^{H \times W \times C_2}$. DPFormer can elegantly cooperate with other feature transformation methods, such as IPM (Mallot et al. 1991), LSS (Philion and Fidler 2020), and so on.

Map element detector. $\mathcal{F}_{\text{BEV}}$ is passed through cascaded transformer decoder layers to generate the fused feature sequences $\mathcal{F}_s$. We adopt hierarchical queries $Q \in \mathbb{R}^{J \cdot M \times C_2}$ designed in MapTR (Liao et al. 2023) to interact with BEV features, where J and M are the maximum numbers of instances and points, respectively. Several sub-heads are designed to predict some basic map element information, as shown in Figure 3 (b).

1) The "Cls Head" outputs a set of semantic categories $\{\hat{c}_j | j = 0, \cdots, J-1\}$ for element instances, where $\hat{c}_j$ are confidence scores for each class (lane dividers, pedestrian



(a) Single-point level (b) Pairwise level (All) (c) Pairwise level (DP)
●: GT DP points ●: GT Aux points ●: PR DP points ●: PR Aux points

Figure 4: Hierarchical constraints on map elements.

crossings, and road boundaries).

2) The "Pts Cls Head" outputs a set $\{\hat{c}'_{j,m}|j=0,\cdots,J-1; m=0,\cdots,M-1\}$ for all points, where $\hat{c}'_{j,m}$ are the confidence scores to indicate whether the point belongs to DP points or auxiliary points.

3) The "Reg Unc Head" outputs not only positions $\hat{v}_{j,m}$, but also uncertainty estimations $\hat{\lambda}_{j,m}$ for all points, indicating the position inference reliability and facilitating the learning process. The predicted set is denoted as $\{(\hat{v}_{j,m}, \hat{\lambda}_{j,m})|j=0,\cdots,J-1; m=0,\cdots,M-1\}$.

Training with Hierarchical Constraints

A map element can be described as a global instance, a collection of local points, or a series of pairwise points. In view of this, we impose constraints at different levels to retain their geometric and semantic properties. The total loss function consists of an instance-level classification loss L_{cls}, a point-level classification loss $L_{\text{pts\_cls}}$, a position with uncertainty loss $L_{\text{pts\_unc}}$, and a pairwise-point shape matching loss L_{s}, formulated as:

$$L_{\text{Total}} = \alpha_{c_1} L_{\text{cls}} + \alpha_{c_2} L_{\text{pts\_cls}} + \alpha_p L_{\text{pts\_unc}} + \alpha_s L_{\text{s}}, \quad (1)$$

where α_{c_1}, α_{c_2}, α_p, and α_s are weight coefficients to balance loss terms.

Instance-level classification loss. Given the instance-level bipartite matching correspondences $\hat{\pi}$, which are optimized using the Hungarian algorithm (Kuhn 1955), the instance-level classification loss can be defined by Focal Loss (Lin et al. 2017):

$$L_{\text{cls}} = \sum_{j=0}^{J-1} L_{\text{Focal}}(c_j, \hat{c}_{\hat{\pi}(j)}), \quad (2)$$

where the j-th ground-truth instance is matched with the $\hat{\pi}(j)$-th prediction.

Point-level classification loss. Besides, the point-level classification loss is adopted to constrain whether each point belongs to the DP point or the auxiliary point, denoted as:

$$L_{\text{pts\_cls}} = \sum_{j=0}^{J-1} \mathbb{1}(c_j \neq 0) \sum_{m=0}^{M-1} L_{\text{Focal}}(c'_{j,\hat{\gamma}_j(m)}, \hat{c}'_{\hat{\pi}(j),m}), \quad (3)$$

where $\mathbb{1}(\cdot)$ is an indicator function, $\hat{\gamma} \in \Gamma$ is the optimal point-level assignment, and the m-th predicted point on the $\hat{\pi}(j)$-th element corresponds to the $\hat{\gamma}_j(m)$-th ground-truth point located on the j-th element.

Position with uncertainty loss. Considering that viewpoint transformations may lead to unreliable ground truths due to calibration errors, space compression, and manual labeling, we design a position with uncertainty loss to promote the reliability and robustness of position estimations. To achieve this goal, we assume the position of each predicted point follows a Laplace distribution $La(\mu, \frac{\sigma}{\sqrt{2}})$, where μ and σ are the mean and standard deviation, respectively.

Similar to the point-level classification loss, we only impose supervision on the matched predictions. σ is used to model the heteroscedastic aleatoric uncertainty in position estimations. Aiming at avoiding a potential division by zero, our network directly predicts $\hat{\lambda}_{j,m} := \log(\sigma_{j,m})$ instead, as (Kendall and Gal 2017) defines. The loss function is formulated as:

$$L_{\text{pts\_unc}} = \sum_{j=0}^{J-1} \mathbb{1}(c_j \neq 0) \sum_{m=0}^{M-1} \Big(\sqrt{2}\exp(-\hat{\lambda}_{\hat{\pi}(j),m}) \cdot D_{\text{Manhattan}}(v_{j,\hat{\gamma}_j(m)}, \hat{v}_{\hat{\pi}(j),m}) + \hat{\lambda}_{\hat{\pi}(j),m}\Big), \quad (4)$$

where $D_{\text{Manhattan}}(v_{j,\hat{\gamma}_j(m)}, \hat{v}_{\hat{\pi}(j),m})$ is Manhattan distance.

Pairwise-point shape matching loss. Pairwise points, as a continuum between global instances and local points, can provide more informative regional shapes. For each map element, pairwise-point shape matching constraints are imposed on shapes connected by adjacent points of V (pink in Figure 4 (b)) and V^{DP} (orange in Figure 4 (c)), respectively. Considering that most map elements only involve 2 DP points, leading to weakly annotated ground truths, the former imposes constraints on all points to introduce more positional and structural information. While the latter constrains the fundamental trend of the overall instances.

The j-th element instance of pairwise points is denoted as S_j, where $|S_j| = M + n_d^j - 2$, and $n_d^j = \sum_{m=0}^{M-1} \mathbb{1}(c'_{j,m} \neq 0)$. Each $s \in S_j$ is described by a set of shape parameters (center positions, length, and radian). We construct a 2D Gaussian distribution $g_s(\mu', \Sigma)$ to encode the chain coupling relationship composed of s, where the mean μ' is made up of center positions, and the covariance Σ is computed by the length and radian. The corresponding Gaussian distribution of the prediction is denoted as $g_{\hat{s}}$, where $\hat{s} \in \hat{S}_{\hat{\pi}(j)}$. Each $g_{\hat{s}}$ is constructed in the same manner as the ground truth, with further details found in (Liu et al. 2023a; Guan et al. 2023). Pairwise-point shape matching loss is designed to utilize the multi-scale structure to help learn a rich and robust feature representation and restrict the parameter space during optimization. The joint distributions of predictions and ground truths are forced together using the symmetric

Algorithm 1: DP point generation.

Input: Annotated points $V^{\dagger}$
Parameters: Maximum point number N, Initial tolerance τ, Tolerance factor ϵ
Output: DP points V^{DP}

$V' = \mathrm{FindStartEnd}(V^{\dagger})$
$V^{\mathrm{DP}} = \mathrm{DPSimplification}(V', \tau)$
while $|V^{\mathrm{DP}}| > N$ **do**
  $\tau = \tau \times \epsilon$
  $V^{\mathrm{DP}} = \mathrm{DPSimplification}(V^{\mathrm{DP}}, \tau)$
end while
return V^{DP}

Kullback-Leibler divergence (KLD):

$$L_{\mathrm{s}} = \frac{1}{2}\sum_{j=0}^{J-1} \mathbb{1}(c_j \neq 0) \sum_{s \in S_j} (D_{\mathrm{kl}}(g_{\hat{s}}, g_s) + D_{\mathrm{kl}}(g_s, g_{\hat{s}})). \tag{5}$$

Training Strategies

Ground-truth DP point generation. Taking original annotated points $V^{\dagger}$ as input, we expect to select a set of DP points for map element compaction. Given the preset maximum point number of N, a small initial tolerance τ, and a reasonable tolerance factor ϵ, DP points are iteratively generated as presented in Algorithm 1. Notably, for closed-shape elements, the FindStartEnd function searches for the two points with the furthest distance and selects either as the starting/ending point. And the DPSimplification function is implemented by the Douglas-Peucker algorithm.

Hierarchical matching. The hierarchical matching, consisting of instance-level matching and point-level matching, can be carried out in unified processing, as (Carion et al. 2020; Yin et al. 2021) proposed. To achieve the instance-level matching, we construct a correspondence $\hat{\pi}$ between J predicted map elements and G ground truths (padded to J). The cost matrix $\mathcal{C} \in \mathbb{R}^{J \times J}$ is designed covering a classification cost $\mathcal{C}_{\mathrm{cls}}$ and a position cost $\mathcal{C}_{\mathrm{pos}}$:

$$\begin{aligned}\hat{\pi} = &\arg\min_{\pi} \sum_{j=0}^{J-1} (\alpha_{c_1} \mathcal{C}_{\mathrm{cls}}(\hat{c}_{\pi(j)}, c_j) \\ &+ \alpha_p \mathcal{C}_{\mathrm{pos}}(\hat{V}_{\pi(j)}, V_{\hat{\gamma}_j})),\end{aligned} \tag{6}$$

where $\mathcal{C}_{\mathrm{cls}}$ is computed at the instance level using Focal loss, $\hat{V}_{\pi(j)}$ is the $\pi(j)$-th predicted point set, and $V_{\hat{\gamma}_j}$ is the best permutation of the j-th ground-truth point set. α_{c_1} and α_p are weight terms set w.r.t. the loss function configuration.

Besides, for the j-th ground-truth instance, $\hat{\gamma}_j$ is obtained by selecting the minimum Manhattan distance of all the possible point permutations Γ_j:

$$\hat{\gamma}_j = \arg\min_{\gamma_j \in \Gamma_j} \sum_{m=0}^{M-1} D_{\mathrm{Manhattan}}(\hat{v}_{j,m}, v_{j,\gamma_j(m)}). \tag{7}$$

Finally, with the help of the Hungarian algorithm, a one-to-one assignment with the minimum cost can be acquired.

Experiments

Dataset and evaluation metrics. Experiments are conducted on the large-scale nuScenes (Caesar et al. 2020) dataset, covering a total of 1000 driving scenes, each sample with 6 camera images and LiDAR sweeps attached. Due to the unavailability of annotations for the testing set, we follow (Li et al. 2022a; Liu et al. 2023b; Liao et al. 2023) to train on the 700 scenes (28130 samples) from the training set and test on the 150 scenes (6019 samples) from the validation set. The perception space covers a range of $[-15\mathrm{m}, 15\mathrm{m}]$ along the x-axis and $[-30\mathrm{m}, 30\mathrm{m}]$ along the y-axis.

To quantify the quality of map constructions, fair evaluations are organized on lane dividers, pedestrian crossings, and road boundaries. We use the Chamfer-distance-based metric to compare the similarity between predictions and ground truths. The average precision (AP) is computed under different thresholds $[0.5, 1.0, 1.5]$m, and mAP is obtained by averaging results across all thresholds. Besides, we report compression ratios $R_{\mathrm{comp}} = \frac{N_{\mathrm{PR}}}{N_{\mathrm{ori\_GT}}}$, where N_{PR} is the total number of predicted points used to represent all map elements in the validation dataset, and $N_{\mathrm{ori\_GT}}$ is the total number of original ground-truth points.

Implementation details. For the backbone, we employ the commonly used ResNet50 (He et al. 2016), with the learning rate multiplier set to 0.1. The viewpoint transformation module is selected as GKT (Chen et al. 2022b). We train DPFormer with a batch size of 4 (each containing $K = 6$ images). The AdamW (Loshchilov and Hutter 2019) optimizer is adopted with a learning rate of $1.25e^{-4}$ for single-card training. According to data statistics shown in Figure 2, we set the maximum number of DP points as $N = 8$, the maximum number of all points as $M = 20$, and the maximum number of map elements as $J = 50$. Besides, the loss coefficients α_{c_1}, α_{c_2}, α_p and α_s are set to 2, 0.5, 5, and 0.5, respectively. All experiments are conducted on a single Nvidia RTX 3090. Ablation studies are trained with 50 epochs.

Comparisons with State-of-the-Art Methods

We take three works with open-sourced codes as our competitors, a segmentation-based method HDMapNet (Li et al. 2022a), a polyline-generated method VectorMapNet (Liu et al. 2023b), and a DETR-like method MapTR (Liao et al. 2023). Comparative results are reported in Table 1. Considering unaligned configurations across different numbers of GPUs, we retrain MapTR on a single RTX 3090 until it achieves results consistent with those reported in the paper. Our training epoch counts are determined based on this to ensure equitable and consistent comparisons. Our DPFormer outperforms the latest MapTR with a 2.1% higher mAP. Besides, we report compression ratios based on the threshold for visualizations. DPFormer and MapTR respectively achieve compression ratios of 46.7% and 205.1%, i.e., DPFormer uses about 4× fewer points than MapTR to facilitate compact HD map construction. Qualitative visualizations presented in Figure 5 indicate that DPFormer effectively preserves the structures and contours of map elements.

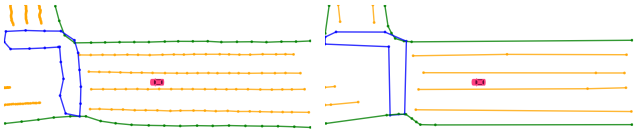
To better show comparisons around topological changes, we utilize ground-truth DP points to identify vital locations

Method	Backbone	N_{GPU}	Epoch	$AP_{divider}$	AP_{ped}	$AP_{boundary}$	mAP	FPS
HDMapNet	Effi-B0	-	30	21.7	14.4	33.0	23.0	0.9
	PointPillars	-	30	24.1	10.4	37.9	24.1	-
	Effi-B0 & PointPillars	-	30	29.6	16.3	46.7	31.0	-
VectorMapNet	R50	8	110	47.3	36.1	39.3	40.9	3.9
	PointPillars	8	110	37.6	25.7	38.6	34.0	-
	R50 & PointPillars	8	110	50.5	37.6	47.5	45.2	-
MapTR	R50	8	24	51.5	46.3	53.1	50.3	13.3
	R50	8	110	59.8	56.2	60.1	58.7	
DPFormer	R50	1	50	57.8	51.3	54.7	54.6	11.3
	R50	1	150	63.2	59.3	60.0	60.8	

Table 1: Comparative results on the nuScenes dataset. "Effi-B0" (Tan and Le 2019), "PointPillars" (Lang et al. 2019), and "R50" (He et al. 2016) are the widely used backbones. Methods with two backbones are based on multi-modal inputs. Our training epoch counts are determined by experiments on MapTR, which is trained on a single RTX 3090 until it reaches the reported performance benchmarks. We retest the FPS for all the methods on a single RTX 3090 for fair comparison.

EP	DP	L_{p2p}	L_{pts_unc}	L_{dir}	L_s	$AP_{divider}$	AP_{ped}	$AP_{boundary}$	mAP
√		√		√		52.3	47.3	55.2	51.6
	√	√		√		56.0	48.6	51.9	52.2
	√		√	√		56.4	50.5	52.9	53.3
	√		√		√	57.8	51.3	54.7	54.6

Table 2: Ablation studies on each component of our DPFormer. The result in the first row is equivalent to that of MapTR.



(a) MapTR (b) DPFormer

Figure 5: Qualitative visualizations of competitive methods.

Metric	Results (2 m)		Results (5 m)	
	MapTR	Ours	MapTR	Ours
$AP_{divider}$	48.3	54.5	53.3	59.8
AP_{ped}	43.2	53.3	44.6	50.9
$AP_{boundary}$	47.1	53.1	50.1	54.6
mAP	46.2	53.6	49.3	55.1

Table 3: Comparative results within the neighborhoods decided by ground-truth DP points.

and evaluate points within the specific neighborhoods (radius). Results are reported in Table 3. DPFormer outperforms MapTR with 7.4% and 5.8% higher mAP within the neighborhoods of 2 m and 5 m. Specifically, DPFormer beats MapTR with 10.1% and 6.3% higher AP_{ped} within the specific neighborhoods, which can be attributed to DP points' capacity of capturing the indispensable structures.

Figure 6 illustrates the visualization results of DPFormer. The upper row demonstrates DPFormer's ability to tackle map elements with diverse semantic information, shapes, and arbitrary directions, enabling point predictions of a flexible number. The lower row showcases the vital role played by our DPFormer in constructing compact HD maps, which involves fewer points to represent map elements, especially in scenarios with simple topology, achieving a compression ratio of 45.5% in the current sample.

Ablation Studies

Extensive ablation studies are conducted to verify the effectiveness of each component, including replacing the equidistant point representation ("EP") with our DP point representation ("DP"), replacing the point2point loss ("L_{p2p}") adopted in (Liao et al. 2023) with our position loss with uncertainty ("L_{pts_unc}"), and replacing the commonly used cosine similarity loss ("L_{dir}") with our pairwise-point shape matching loss ("L_s"). Results are reported in Table 2.

Effectiveness of representation with DP points. Compared to the equidistant point representation used in MapTR, our DP point representation involves more essential points to depict map elements, thereby achieving competitive increases of 3.7% and 1.3% in AP for dividers and pedestrian crossings. The reason lies in their inherent structural stability, as marked in yellow and blue in Figure 6, which has superiority in mitigating unnecessary regression errors through the utilization of the compact DP points. By comparison, boundaries exhibit more complex and varied appearances in terms of shape openness and topological structures. The sacrifice in accuracy during ground-truth DP point generation diminishes the advantages of DP points. Besides, based on dataset statistics, over half of the map elements (52.8%) can be accurately represented using only 2 points, i.e., the majority of elements are composed of straight lines. Consequently, DP point representation holds promise for compression.

Contribution of position with uncertainty constraint. Compared to L_{p2p} imposed solely on positions of points, our

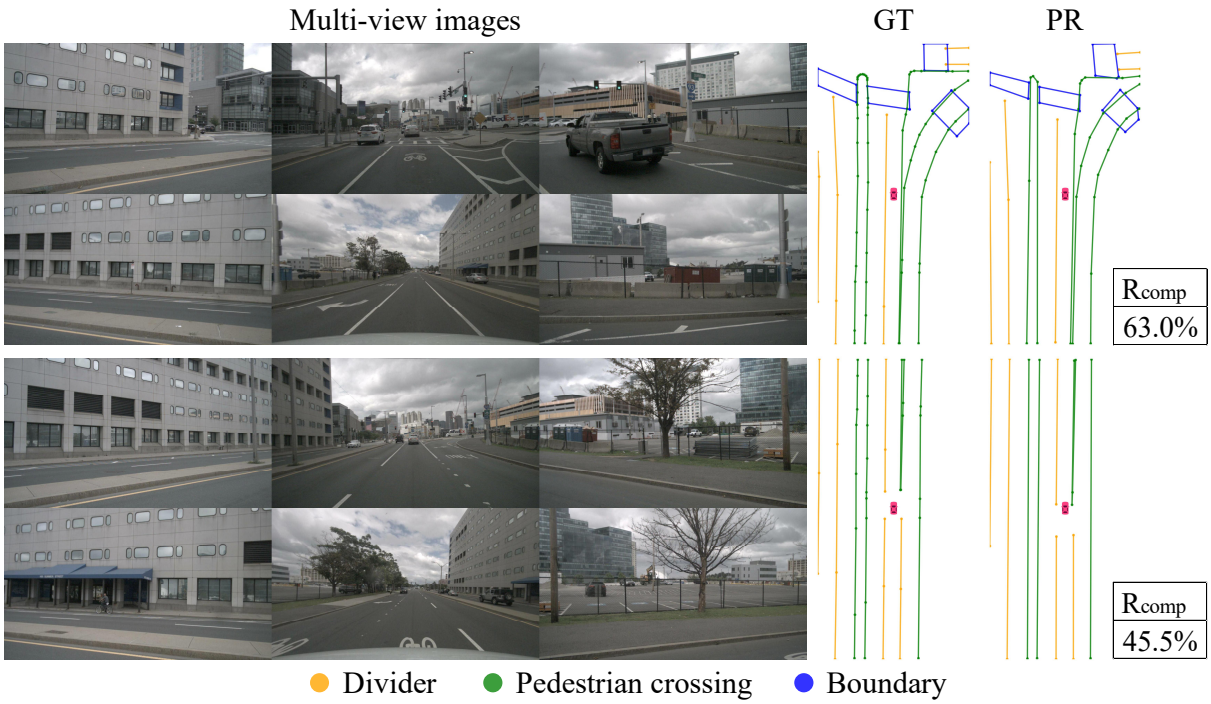


Figure 6: Visualizations of compact HD map construction provided by our DPFormer.

α_p	$AP_{divider}$	AP_{ped}	$AP_{boundary}$	mAP
4.0	54.7	46.3	55.9	52.3
5.0	57.8	51.3	54.7	54.6
6.0	55.2	49.6	54.4	53.1

Table 4: Ablations on position with uncertainty constraint.

α_s	$AP_{divider}$	AP_{ped}	$AP_{boundary}$	mAP
0	53.2	49.4	51.4	51.3
0.25	56.7	48.2	51.7	52.2
0.5	57.8	51.3	54.7	54.6
0.75	56.9	49.6	51.5	52.7

Table 5: Ablations on pairwise-point shape matching constraint.

$L_{\text{pts\_unc}}$ introduces an uncertainty estimation, thereby relaxing the stringent requirements on position regression in early training, and resulting in a 1.1% improvement in mAP. We note that the improvement on uncertainty associated with complex shapes is higher than that of simple shapes. Besides, α_p are set to different weight to further validate the importance of $L_{\text{pts\_unc}}$, as shown in the Table 4.

Efficacy of pairwise-point shape matching constraints. In Table 5, we ablate on the pairwise-point shape matching constraint, with α_s set as $[0, 0.25, 0.5, 0.75]$, where $\alpha_s = 0$ means we do not use the pairwise-point shape matching constraint. With α_s increasing, the pairwise-point shape matching constraints can impose an incremental force on the coupled relationships containing both directions and positions, relieving the local errors brought by isolated point regression. However, when α_s is too large, overmuch attention on shapes leads to ambiguity in positions.

Conclusion

In this paper, we propose DPFormer, an end-to-end framework for compact HD map construction. DPFormer employs an effective transformer-based architecture for the flexible detection of class-agnostic map elements. In particular, a compact Douglas-Peucker point representation is proposed to achieve precise element approximation, and the related constraints with uncertainty estimation as well as shape matching strategies are designed accordingly for robust and reliable results. Experiments on the challenging nuScenes dataset validate that DPFormer has superiority in precision. In future work, we will combine DPFormer with other auxiliary tasks for better performances, and extend it to downstream tasks, such as multi-modality predictions and motion planning.

Acknowledgements

This work was supported by the National Key R&D Program of China (2023YFB4704900) and the National Natural Science Foundation of China (61976170, 62088102).

References

Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2020. nuScenes: A Multimodal Dataset for Autonomous

Driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Cao, Z.; Li, A.; Xiong, Z.; and Yuan, Z. 2021. Surround-view Free Space Boundary Detection with Polar Representation. In *Proceedings of the British Machine Vision Conference (BMVC)*.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-End Object Detection with Transformers. In *European Conference on Computer Vision (ECCV)*.

Chen, L.; Sima, C.; Li, Y.; Zheng, Z.; Xu, J.; Geng, X.; Li, H.; He, C.; Shi, J.; Qiao, Y.; et al. 2022a. Persformer: 3D Lane Detection via Perspective Transformer and the Openlane Benchmark. In *European Conference on Computer Vision (ECCV)*.

Chen, S.; Cheng, T.; Wang, X.; Meng, W.; Zhang, Q.; and Liu, W. 2022b. Efficient and Robust 2D-to-BEV Representation Learning via Geometry-guided Kernel Transformer. *ArXiv Preprint arXiv:2206.04584*.

Ding, W.; Qiao, L.; Qiu, X.; and Zhang, C. 2023. PivotNet: Vectorized Pivot Learning for End-to-end HD Map Construction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Douglas, D. H.; and Peucker, T. K. 1973. Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or Its Caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*.

Feng, Z.; Guo, S.; Tan, X.; Xu, K.; Wang, M.; and Ma, L. 2022. Rethinking Efficient Lane Detection via Curve Modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Guan, Z.; Liu, R.; Yuan, Z.; Liu, A.; Tang, K.; Zhou, T.; Li, E.; Zheng, C.; and Mei, S. 2023. Flexible 3D Lane Detection by Hierarchical Shape Matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kendall, A.; and Gal, Y. 2017. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? *Advances in Neural Information Processing Systems (NIPS)*.

Kuhn, H. W. 1955. The Hungarian Method for the Assignment Problem. *Naval Research Logistics Quarterly*.

Lang, A. H.; Vora, S.; Caesar, H.; Zhou, L.; Yang, J.; and Beijbom, O. 2019. PointPillars: Fast Encoders for Object Detection from Point Clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Li, Q.; Wang, Y.; Wang, Y.; and Zhao, H. 2022a. HDMapNet: An Online HD Map Construction and Evaluation Framework. In *International Conference on Robotics and Automation (ICRA)*.

Li, Z.; Wang, W.; Li, H.; Xie, E.; Sima, C.; Lu, T.; Qiao, Y.; and Dai, J. 2022b. BEVFormer: Learning Bird's-Eye-View Representation from Multi-Camera Images via Spatiotemporal Transformers. In *European Conference on Computer Vision (ECCV)*.

Liao, B.; Chen, S.; Wang, X.; Cheng, T.; Zhang, Q.; Liu, W.; and Huang, C. 2023. MapTR: Structured Modeling and Learning for Online Vectorized HD Map Construction. In *International Conference on Learning Representations (ICLR)*.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollar, P. 2017. Focal Loss for Dense Object Detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Liu, L.; Chen, X.; Zhu, S.; and Tan, P. 2021a. CondLaneNet: A Top-To-Down Lane Detection Framework Based on Conditional Convolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Liu, R.; Chen, D.; Liu, T.; Xiong, Z.; and Yuan, Z. 2022. Learning to Predict 3D Lane Shape and Camera Pose from a Single Image via Geometry Constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Liu, R.; Guan, Z.; Yuan, Z.; Liu, A.; Zhou, T.; Kun, T.; Li, E.; Zheng, C.; and Mei, S. 2023a. Learning To Detect 3D Lanes by Shape Matching and Embedding. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*.

Liu, R.; Yuan, Z.; Liu, T.; and Xiong, Z. 2021b. End-to-End Lane Shape Prediction With Transformers. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*.

Liu, Y.; Yuan, T.; Wang, Y.; Wang, Y.; and Zhao, H. 2023b. VectorMapNet: End-to-end Vectorized HD Map Learning. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*.

Loshchilov, I.; and Hutter, F. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations (ICLR)*.

Mallot, H. A.; Bülthoff, H. H.; Little, J.; and Bohrer, S. 1991. Inverse Perspective Mapping Simplifies Optical Flow Computation and Obstacle Detection. *Biological cybernetics*.

Pan, Y.; Xiao, P.; He, Y.; Shao, Z.; and Li, Z. 2021. MULLS: Versatile LiDAR SLAM via Multi-metric Linear Least Square. In *IEEE International Conference on Robotics and Automation (ICRA)*.

Peng, L.; Chen, Z.; Fu, Z.; Liang, P.; and Cheng, E. 2023. BEVSegFormer: Bird's Eye View Semantic Segmentation From Arbitrary Camera Rigs. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*.

Philion, J.; and Fidler, S. 2020. Lift, Splat, Shoot: Encoding Images from Arbitrary Camera Rigs by Implicitly Unprojecting to 3D. In *European Conference on Computer Vision (ECCV)*.

Qiao, L.; Ding, W.; Qiu, X.; and Zhang, C. 2023. End-to-End Vectorized HD-Map Construction With Piecewise Bezier Curve. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Shan, T.; and Englot, B. 2018. LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Shan, T.; Englot, B.; Meyers, D.; Wang, W.; Ratti, C.; and Rus, D. 2020. LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Tabelini, L.; Berriel, R.; Paixao, T. M.; Badue, C.; De Souza, A. F.; and Oliveira-Santos, T. 2021. Keep Your Eyes on the Lane: Real-Time Attention-Guided Lane Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Tan, M.; and Le, Q. 2019. Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks. In *International conference on machine learning (ICML)*.

Yin, Z.; Liu, R.; Yuan, Z.; and Xiong, Z. 2021. Order-Independent Matching with Shape Similarity for Parking Slot Detection. In *Proceedings of the British Machine Vision Conference (BMVC)*.