TEAR of the SUNSET: A Benchmark for Road Detection in Off-Road Environment

Haonan Xu[1,2], Qirui Hu[1,2], Xinyuan Liu[1,2], Hu Li[1,2], Hang Xu[3], Yike Ma[† 1], Yucheng Zhang[1], Feng Dai[1]

[1]*Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China*

[2]*University of Chinese Academy of Sciences, Beijing, China*

[3]*Hangzhou Dianzi University, Zhejiang, China*

{xuhaonan23s, huqirui24s, liuxinyuan21s, lihu21s2, ykma, zhangyucheng, fdai}@ict.ac.cn, hxu@hdu.edu.cn

***Abstract*—Recent advances in autonomous driving technology have enabled its mature deployment in structured urban scenarios that rely on standardized artificial markers for perception. However, this reliance limits its applicability in unstructured environments. Semi-structured environments are a subset of unstructured environments characterized by the absence of artificial road markings but the presence of road traces. To address road detection in such environments, this work proposes a dedicated benchmark. We design a novel road representation that models the road edge lines using higher-order Bézier curves. Meanwhile, we construct the annotated SUNSET dataset, tailored for road detection tasks in such environments. Furthermore, we present TEAR, an end-to-end road detection method which employs an Interconvertible Dual-Instance decoder to decouple road and line instances. We also design a Hierarchical Bipartite Matching strategy for instance association. The experimental results demonstrate that our method achieves excellent performance on the proposed benchmark. Code is available at https://github.com/DoubleFlicker7/TEAR-of-SUNSET.**

***Index Terms*—Benchmark, Road Detection, Semi-Structured Environments**

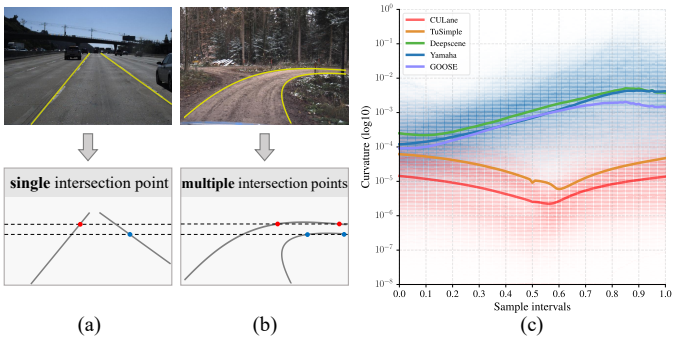

Fig. 1. (a) and (b) demonstrate significant geometric differences in road morphologies between structured environments and semi-structured environments. For the former, lane markings strictly adhere to a single valued correspondence in the Cartesian coordinate. For the latter, road edge lines with large curvatures exhibit multi-valued correspondences. (c) presents the logarithmic curvature distribution at each line sampling point for **traffic environments** (CULane and TuSimple) and **semi-structured environments** (SUNSET) in the form of a heatmap. Solid lines of different colors represent the average curvature of each dataset at each line sampling point.

I. Introduction

Recently, research on autonomous driving technology has achieved remarkable progress and achieved mature deployment in various urban scenarios. Various types of sensors assist vehicles in path planning and decision execution by detecting and recognizing pre-defined lane markings, traffic lights, and other artificial signs in the surrounding environment [1], [2]. However, most real-world environments lack such highly standardized artificial markings, rendering current autonomous driving technology incapable of handling these regions characterized by complexity and unpredictability. Such environments are defined as **unstructured environments** [3].

Autonomous driving in these environments can address the demands of diverse fields such as agriculture, mining, and search and rescue, thus holding substantial practical significance. Road perception tasks in unstructured environments are typically addressed as classic semantic segmentation tasks. However, at intersections, the segmentation results of multiple roads tend to overlap, making it impossible for subsequent modules to effectively decouple road-level topological information from the segmentation outputs. In daily life, we frequently encounter scenarios that lack clear artificial markings but contain ***road traces*** left by human activities. Drivable areas in such scenarios can be identified through explicit cues (e.g., tire tracks, traces of repeated human passage, low-standard roads constructed by humans). We define such environments as **semi-structured environments**. Based on the characteristics of semi-structured environments, we propose a lane-like detection benchmark, aiming to provide lane-like topological information for road perception in semi-structured scenarios and contribute a novel road detection paradigm.

We propose to represent the edge lines of each individual road instance in this environment using a set of Bézier curve pairs, and to define the road area as the geometric region enclosed by these curve pairs. Work [4] first adopted cubic curves to model lane markings in traffic scenarios. However, as is shown in Figure 1, through mathematical statistics, we find that the curvature distribution of roads in semi-structured environments is significantly wider than that of lane markings in structured traffic scenarios. Therefore, we employ higher-order Bézier curves to characterize the contour of roads in semi-

This work was supported by the Science and Technology Project of the Ministry of Agriculture and Rural Affairs of China, and National Natural Science Foundation of China (62372433).

[†]Corresponding Author

structured environments. The curve-based road representation method not only encodes high-level semantic road topological information, enabling direct output of road instance-level results without downstream post-processing, but also remains compatible with the road perception pipeline in structured environments. Moreover, it facilitates the migration of mature existing road detection algorithms to semi-structured environments. Based on the aforementioned data structure, we present a dataset named **SUNSET** (**S**elected **U**nion of **N**umerous datasets for **S**emi-structured **E**nvironments with **T**rails) for road detection tasks in semi-structured environments. The raw images of SUNSET are sourced from three representative outdoor datasets: DeepScene [5], YCOR [6], and GOOSE [7]. These images cover all seasons and various weather conditions (e.g., sunny, rainy, snowy days), and include diverse typical field scenarios such as forests, farms, grasslands, and mining areas. Then, we utilize the Labelme enhanced with a curve annotation function to add curve annotations to these images.

On this basis, we propose an end-to-end road detection method called **TEAR** (**T**wo **E**dge lines define **A** **R**oad) suitable for this task, along with a set of metrics to evaluate the detection performance. We design an **Interconvertible Dual-Instance** Transformer decoder to implicitly model the relationship between road instances and line instances in the query space. We also propose a **Hierarchical Bipartite Matching** method to assign the closest ground truth sample to each group of predicted road samples in sample matching. In the experimental phase, we compare TEAR with several mainstream lane detection algorithms on this benchmark, and achieve promising experimental results.

In summary, we propose a novel benchmark for road detection in semi-structured environments, which mainly consists of the following four components:

- **Benchmark Dataset.** We introduce the SUNSET dataset targeting road detection in semi-structured environments, and provide annotations that conform to the topological logic of roads.
- **Benchmark Task.** We define the concept of *semi-structured environments*, design a road representation method with lane-level information to model road regions in such environments, and propose a novel road detection task tailored for this scenario.
- **Evaluation Metrics.** We propose using *line-level metric* $F1_E$ and *road-level metric* $F1_R$ to evaluate the performance of different methods on this benchmark.
- **Reference Baseline.** We propose an end-to-end road detection method called **TEAR**, which achieves competitive performance on our proposed benchmark, outperforming other lane detection methods by approximately **10%-20%** in the $F1_E$ metric.

II. Methodology

A. Task Definition

For classic object detection tasks, the goal is to identify and locate objects of interest within a given image [8]. Similarly, our road detection task aims to identify and localize road regions by predicting a set of road edge line parameters. We represent road elements through two instance-level forms. **1) Line Instance:** The edge line of the road is abstracted into an n-th order Bézier curve with n+1 control points in Equation 1, which means the number of predicted parameters is $2(n+1)$.

$$\begin{aligned} E(t) &= \sum_{i=0}^{n} b_{i,n}(t)\mathcal{P}_i, 0 \le t \le 1, \\ b_{i,n} &= C_n^i t^i (1-t)^{n-i}, i = 0, \ldots, n \end{aligned} \tag{1}$$

Here, $\mathcal{P}_i$ denotes the i-th control point, $b_{i,n}$ represents the n-th degree Bernstein basis polynomial. The left and right edge lines of the i-th road are denoted as $E^l(t)$ and $E^r(t)$ respectively. Furthermore, the order of control points in the parametric curve effectively reflects the direction of the road in the real world. Therefore, the model outputs the coordinates of the curve's control points in a continuous sequence. **2) Road Instance:** A road instance is represented as a continuous mask region $\Omega(s,t)$, whose shape is determined by the enclosed area of two line instances under artificially defined rules. The relationship between line instances and road instances is described in terms 2.

$$\Omega(u,t) = uE^l(t) + (1-u)E^r(t), t \in [0,1], u \in [0,1] \tag{2}$$

B. Interconvertible Dual-Instance Decoder

Due to the complexity of semi-structured environments, we argue that it is challenging for the network to directly predict the position of the curves in image. Therefore, the **I**nterconvertible **D**ual-**I**nstance (I-DI) decoder enables the network to first learn road features and then model the relationship between road and edge lines on this foundation, facilitating knowledge transfer between two types of instances via two learnable instance converters. The road queries Q_R first pass through a self-attention layer, which allows the queries to exchange information with each other, and a cross-attention layer to perceive the position of the road in the global image. Subsequently, each Q_R simultaneously passes through two **road-to-line instance converters** ($\mathcal{T}_{RE}^l$ and $\mathcal{T}_{RE}^r$) with different parameters, splitting into two edge line queries Q_E^l and Q_E^r (Equation 3). These line queries then perform cross-attention computation with the encoder output $\hat{\mathcal{M}}$ respectively. The computed Q_E^l and Q_E^r are then concatenated along the feature dimension and converted back into $\bar{Q}_R$ via a **line-to-road instance converter** ($\mathcal{T}_{ER}$, Equation 4). These instance converters are essentially feed-forward networks and share the same parameters among all decoder layers, for the mutual conversion between Q_R and Q_E is a deterministic process. Through this process, the Q_R complete a full round of dual-instance conversion from $Q_R \rightarrow Q_E \rightarrow \bar{Q}_R$. Lastly, the queries pass through a classic FFN layer to obtain road-level query outputs.

$$Q_E^l = \mathcal{T}_{RE}^l(Q_R); \quad Q_E^r = \mathcal{T}_{RE}^r(Q_R) \tag{3}$$

$$\bar{Q}_R = \mathcal{T}_{ER}(\text{Concat}(Q_E^l, Q_E^r)) \tag{4}$$

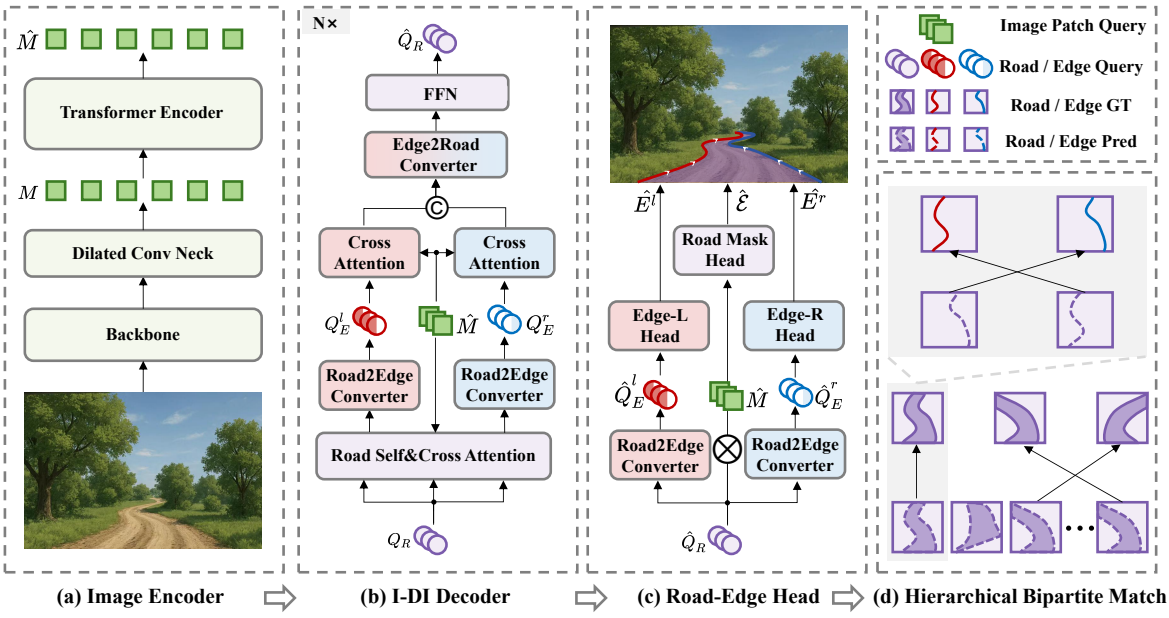

Fig. 2. Overview of our Method. The predicted image undergoes feature extraction via the backbone network and global feature interaction through the Transformer encoder. In each layer of the decoder, road instance queries and line instance queries undergo a round of mutual conversion. Meanwhile, multi-head cross-attention are performed between these queries and the $\hat{\mathcal{M}}$ at both line and road levels. Finally, each road instance query output by the decoder is split into two line instance queries via two trained *Road2Edge Converters*. During the Hierarchical Bipartite Matching phase, each pair of line instances first completes line-level matching, followed by road-level matching, so as to assign the optimal line ground truth to each predicted road edge line.

where $Q_R \in \mathbb{R}^{N_R \times d}$, $\bar{Q}_R \in \mathbb{R}^{N_R \times d}$, $Q_E^l \in \mathbb{R}^{N_E \times d}$, $Q_E^r \in \mathbb{R}^{N_E \times d}$, N_R denotes the number of Q_R, N_E denotes the number of Q_E, and d denotes the feature dimension, $N_R = N_E = 20$ in our method. The input dimension of $\mathcal{T}_{RE}^l$, $\mathcal{T}_{RE}^r$, $\mathcal{T}_{ER}$ is d, d and $2d$, while the output dimension of all converters is d equally.

C. Hierarchical Bipartite Match

Matching Procedure. Since the predicted samples contain hierarchical semantics, we propose to use Hierarchical Bipartite Match to match these predicted samples with the most appropriate ground truth. Define z as the ground truth set of road instances, $\hat{\boldsymbol{z}} = \{\hat{z}_i\}_{i=1}^{N_R}$ as the set of N_R road predictions. For line instance, define y as the set of ground truth, $\hat{\boldsymbol{y}} = \{\{\hat{y}_j^{(i)}\}_{j=1}^{2}\}_{i=1}^{N_R}$ as the set of $2N_R$ line predictions, the relationship between $\hat{z}$ and $\hat{y}$ follows $\mathbb{T}$: $\hat{z}_i \Leftrightarrow \{\hat{y}_1^{(i)}, \hat{y}_2^{(i)}\}$ (so do z and y). Our final goal is to find the best bipartite matching between two road-level sets:

$$\hat{\sigma}_R = \underset{\sigma_R \in \mathfrak{S}_{N_R}}{\arg\min} \sum_{i=1}^{N_R} \mathcal{C}_{\text{Rmatch}}\left(z_i, \hat{z}_{\sigma_R(i)}\right) \tag{5}$$

To calculate $\mathcal{C}_{\text{Rmatch}}\left(z_i, \hat{z}_{\sigma_R(i)}\right)$, we then need to find the best bipartite matching between two line-level sets from z_i and $\hat{z}_{\sigma_R(i)}$, and after applying $\mathbb{T}$, the $\left(z_i, \hat{z}_{\sigma_R(i)}\right)$ can be represent as $\left(\{y_1^{(i)}, y_2^{(i)}\}, \{\hat{y}_1^{\sigma_R(i)}, \hat{y}_2^{\sigma_R(i)}\}\right)$. In line-level bipartite matching, there are only two elements in permutation, so we represent it in follow form:

$$\mathfrak{S}_2^{(i)} = \begin{Bmatrix} \{\, y_1^{(i)} \Leftrightarrow \hat{y}_1^{\sigma_R(i)},\ y_2^{(i)} \Leftrightarrow \hat{y}_2^{\sigma_R(i)} \,\}, \\ \{\, y_1^{(i)} \Leftrightarrow \hat{y}_2^{\sigma_R(i)},\ y_2^{(i)} \Leftrightarrow \hat{y}_1^{\sigma_R(i)} \,\} \end{Bmatrix} \tag{6}$$

Considering the universality of line-level bipartite matching, the variable i in the formula can be discard. The line-level bipartite matching is essentially the minimum element value of a binary set, which can be represent in Equation 7. Hierarchical Bipartite Match will eventually obtain an optimal road-level mapping $\hat{\sigma}_R$ (Equation 5) and a set of corresponding optimal line-level mappings $\hat{\pi}_E = \{\hat{\pi}_E^{(i)}\}_{i=1}^{N_R}$.

$$\hat{\pi}_E = \underset{\pi_E \in \mathfrak{S}_2}{\arg\min} \underbrace{\sum_{j=1}^{2} \mathcal{C}_{E\text{match}}\left(y_j, \hat{y}_{\pi_E(j)}\right)}_{\mathcal{C}_{R\text{match}}\left(z_i, \hat{z}_{\sigma_R(i)}\right)} \tag{7}$$

Matching Cost. The matching cost consists of three parts. **1) Confidence Cost.** For the prediction of curve line $\hat{y}_{\hat{\pi}_E^{(i)}(j)}$, we define probability of existence as $\hat{p}_{\hat{\pi}_E^{(i)}(j)}$ ranging from 0 to 1, and the confidence cost is expressed as $\mathcal{C}_{cls} = -\hat{p}_{\hat{\pi}_E^{(i)}(j)}$. **2) Curve Distance Cost.** The parametric curve allows us to precisely calculate the point coordinates at a set (T) of t values. We calculate the L_1 distance at the same sampling value for the two curves and take the arithmetic mean as the distance between two curves, which is the same as that in [4], [9], [10]. The curve distance cost is expressed as $\mathcal{C}_{reg}^{sp} = \frac{1}{N_p}\sum_{t\in T}\|E(t) - \hat{E}(t)\|_1$, where N_p is the number of all sample points. **3) Control points Distance Cost.** Through experiments, we find that relying solely on the curve distance cost is insufficient to achieve accurate localization of higher-order curves. To address this issue, we propose that while using the average curve distance cost, the average L_1 distance between each pair of control points should also be considered. This joint cost calculation strategy ensures the model outputs

control point coordinates in a more reasonable and logically consistent manner, thereby enhancing the localization accuracy and curvature continuity. The control points distance cost can be expressed as $\mathcal{C}_{reg}^{kp} = \frac{1}{R+1}\sum_{k\in K}\|\mathcal{P}(k) - \hat{\mathcal{P}}(k)\|_1$, where $K = \{0, 1, ..., R\}$, R denotes the order of the curve, $\mathcal{P}$ and $\hat{\mathcal{P}}$ are sets of control points for y and $\hat{y}$. The final cost function is shown in Equation 8, where $\alpha_1, \alpha_2, \alpha_3$ are set $1.0, 5.0, 2.0$ respectively.

$$\mathcal{C}_{E\text{match}} = \alpha_1\mathcal{C}_{cls} + \alpha_2\mathcal{C}_{reg}^{sp} + \alpha_3\mathcal{C}_{reg}^{kp} \tag{8}$$

D. Loss Function

The loss function is similar to the cost function. For the matched positive sample pairs, we simultaneously calculate two components as the regression loss for the curves: the L_1 curve distance loss $\mathcal{L}_{reg}^{sp}$ between curve pairs and the L_1 control point distance loss $\mathcal{L}_{reg}^{kp}$ between corresponding control point pairs. For classification loss, we adopt a weighted binary cross-entropy function in Equation 9 to calculate the loss of class confidence scores for all samples. This strategy is intended to mitigate the adverse impact caused by the imbalance in the number of positive and negative samples.

$$\mathcal{L}_{cls} = -\big(w_p * y\log(\hat{p}_{\hat{\pi}_E}) + (1-y)\log(1-\hat{p}_{\hat{\pi}_E})\big) \tag{9}$$

In addition, to enhance the model's ability to understand road instances, we introduce a road-level auxiliary mask loss, which would be removed in inference phase. We use Equation 2 to generate binary mask regions of each road from line annotations and discretize these regions at $H \times W$ resolution:

$$\Omega : \{\Omega_{(u,t)}^{(i)}\}_{i=1}^{N_R^{gt}} \xrightarrow{\text{discrete}} \Pi : \{m_i \mid m_i \in \{0,1\}^{H\times W}\}_{i=1}^{N_R^{gt}} \tag{10}$$

where N_R^{gt} denotes the number of road ground truth instances in one image. Meanwhile, we obtain each road mask prediction $\hat{\mathcal{E}}$ by a dot product between $\hat{\mathcal{M}}$ and $\hat{Q}_R$ along the feature dimension (Equation 11), and then reshape it to the same size as the feature of the last layer of the backbone. To compute the mask loss between Π and $\hat{\mathcal{E}}$, it is necessary to reshape both into the same dimension. We scale Π by a factor of $\frac{1}{\sqrt{s}}$ and $\hat{\mathcal{E}}$ by a factor of $\sqrt{s}$ ($\Pi \in \mathbb{R}^{\frac{H}{\sqrt{s}}\times\frac{W}{\sqrt{s}}\times N_R^{gt}}, \hat{\mathcal{E}} \in \mathbb{R}^{\frac{H}{\sqrt{s}}\times\frac{W}{\sqrt{s}}\times N_R}$), respectively, to minimize the information loss incurred during this process. Then, the auxiliary mask loss is denoted as Equation 12.

$$\hat{\mathcal{E}} = \text{reshape}\left(\hat{\mathcal{M}}\cdot\hat{Q}_R\right), \hat{\mathcal{E}} \in \mathbb{R}^{\frac{H}{s}\times\frac{W}{s}\times N_R} \tag{11}$$

$$\begin{aligned}\mathcal{L}_{mask} = -\frac{s}{HW}\sum_p\sum_q \big(w_m * m_i^{(p,q)}\log\hat{\mathcal{E}}_{\hat{\sigma}_R(i)}^{(p,q)} + \\ (1-m_i^{(p,q)})\log(1-\hat{\mathcal{E}}_{\hat{\sigma}_R(i)}^{(p,q)})\big)\end{aligned} \tag{12}$$

The complete loss of our method denotes in Equation 13, where super-parameter $\lambda_1, \lambda_2, \lambda_3, \lambda_4, w_p, w_m$ are set to 1.0, 5.0, 2.0, 5.0, 10.0 and 5.0.

$$\mathcal{L} = \lambda_1\mathcal{L}_{cls} + \lambda_2\mathcal{L}_{reg}^{sp} + \lambda_3\mathcal{L}_{reg}^{kp} + \lambda_4\mathcal{L}_{mask} \tag{13}$$

III. Experiments

A. Datasets

Work [17] quantitatively compared publicly available datasets in unstructured outdoor environments, from which we selected three field datasets suitable for our task: DeepScene, YCOR, and GOOSE. We extracted images that meet the definition of semi-structured scenarios from these datasets to form dataset. To enhance the scenario diversity of the benchmark, we combined the above three datasets to construct the SUNSET dataset. All methods are also evaluated on this comprehensive dataset.

B. Evaluation Metrics

Line-level. We adopt the official **F1 score** from the lane detection task on the CULane [11] and LLAMAS [18] datasets as the line-level metric (represented as $F1_E$). Each road edge is included in the calculation as an individual line instance. Each curve has a width of 30 pixels, a predicted sample is considered a successful match if its confidence score exceeds 0.9 and its IoU with the ground truth sample exceeds 0.5.

Road-level. The line-level metric only characterizes the detection performance of individual road edge lines, but does not take into account the performance at the road level. In our benchmark, a single road is represented by two parametric curve instances. Therefore, we propose a road-level $F1_R$ metric to evaluate the overall road detection performance. Specifically, the IoU of a road instance is calculated as the average IoU of the two edge lines after line-level matching.

Implementation. We develop these two metrics based on the COCO [19] metric in the object detection task, and all the comparative experiments were evaluated using these metrics.

C. Comparisons

Overview. The overall experimental results are presented in Table I. We selected several representative detection methods based on different paradigms in the field of 2D lane detection over recent years for comparative evaluation [20]. We reimplement all the compared methods based on our custom codebase. Considering that some methods require road existence labels and the corresponding segmentation ground truth, we supplement these pieces of information following the format of the CULane dataset. All methods are trained on the SUNSET training set and evaluated on each of the sub-datasets as well as the SUNSET test set. In addition, we counted the parameters of each model and the FLOPs under the same input image size (512×1024), using these two metrics to assess the real-time performance of the models. Since the concept of road instances is not involved in the lane detection task, we only computed the $F1_E$ metric for all lane detection methods.

Comparison with Detection-based Methods. Our method adheres to the curve-based modeling paradigm. Among all methods of this category, TEAR substantially outperforms the compared approaches across all datasets, while its parameter count is nearly half that of these methods. For BezierLaneNet and LSTR, both of which adopt cubic parametric curves to model lane markings, most road curves in semi-structured

TABLE I
COMPARISON WITH TYPICAL LANE DETECTION METHODS ON SUNSET.

	Methods	Publication	Backbone	DeepScene		YCOR		GOOSE		SUNSET		Para(M)	FLOPs(G)
				$F1_E$	$F1_R$	$F1_E$	$F1_R$	$F1_E$	$F1_R$	$F1_E$	$F1_R$		
♣Segmentation	▼ ***Mask based***												
	SCNN [11]	*AAAI'18*	VGG16	52.26	-	31.75	-	30.67	-	43.72	-	23.73	183.54
	RESA [9]	*AAAI'21*	R-34	56.29	-	33.89	-	34.41	-	48.51	-	12.70	114.01
	▼ ***Keypoints based***												
	GANet [12]	*CVPR'22*	R-18	58.44	-	37.87	-	34.33	-	50.69	-	30.66	30.92
	CondLSTR [13]	*ICCV'23*	R-34	54.10	-	31.92	-	36.41	-	47.18	-	26.88	24.40
♣Detection	▼ ***Line-Anchor based***												
	SRLane [14]	*AAAI'24*	R-18	60.75	-	37.57	-	34.67	-	52.16	-	11.56	18.97
	LaneATT [15]	*CVPR'21*	R-18	51.31	-	38.48	-	32.51	-	45.57	-	23.01	42.73
	▼ ***Curve based***												
	BezierLaneNet [4]	*CVPR'22*	R-34	29.53	-	23.60	-	19.36	-	29.03	-	9.05	32.31
	LSTR [16]	*WACV'21*	R-18	48.15	-	31.07	-	30.52	-	40.56	-	21.94	38.18
	▼ ***Ours***												
	TEAR-S (ResNet18+E2/D2)	-	R-18	68.43	69.57	38.31	33.85	35.89	33.66	59.60	59.81	5.64	19.12
	TEAR-M (ResNet34+E3/D3)	-	R-34	**70.51**	73.07	**39.63**	40.29	**36.64**	34.74	**61.27**	62.56	11.88	38.09

TABLE II
ABLATION STUDY ON DIFFERENT NUMBERS OF TRANSFORMER BLOCKS

Blocks	DeepScene		YCOR		GOOSE		SUNSET	
	$F1_E$	$F1_R$	$F1_E$	$F1_R$	$F1_E$	$F1_R$	$F1_E$	$F1_R$
4	68.88	70.92	39.74	36.81	35.47	33.58	60.32	61.03
3	68.28	70.01	39.01	35.96	36.50	35.85	59.85	60.58
2	68.43	69.57	38.31	33.85	35.89	33.66	59.60	59.81

TABLE III
ABLATION STUDY ON LOSS

$\mathcal{L}_{reg}^{kp}$	$\mathcal{L}_{mask}$	DeepScene		YCOR		GOOSE		SUNSET	
		$F1_E$	$F1_R$	$F1_E$	$F1_R$	$F1_E$	$F1_R$	$F1_E$	$F1_R$
		60.38	64.09	34.21	23.22	31.95	31.09	52.82	54.44
✓		67.47	68.67	34.20	26.67	34.65	31.67	57.97	57.22
	✓	64.76	65.98	35.56	27.83	34.71	32.24	57.15	57.46
✓	✓	68.43	69.57	38.31	33.85	35.89	33.66	59.60	59.81

TABLE IV
ABLATION STUDY ON HIERARCHICAL DECODER MODULES

I-DI	Line MHA	DeepScene		YCOR		GOOSE		SUNSET	
		$F1_E$	$F1_R$	$F1_E$	$F1_R$	$F1_E$	$F1_R$	$F1_E$	$F1_R$
	-	64.95	67.20	33.95	27.11	33.05	32.03	56.60	57.83
✓		65.27	68.00	36.09	30.30	33.55	32.62	57.53	58.52
✓	✓	68.43	69.57	38.31	33.85	35.89	33.66	59.60	59.81

scenarios contain multiple local segments with large curvature, which cannot be effectively fitted by cubic curves. Such scenario bias renders these curve-based detection methods directly ineffective. For lane-anchor-based detection methods, TEAR surpasses them by a margin of **7%–14%** on the SUNSET dataset. Lane-anchor based methods model lane markings using a set of coordinate points sampled at equal intervals along the y-axis. The prerequisite for this method is that there cannot be multiple predictable x-values corresponding to a single sampling point on the y-axis, a constraint that does not hold in semi-structured scenarios.

Comparison with Segmentation-based Methods. Mask-based methods entail dense prediction for each pixel in the image, which consequently results in higher FLOPs. Nevertheless, this paradigm is free from the scenario bias inherent in detection-based approaches, thus endowing it with an inherent advantage in predicting high-curvature roads in semi-structured scenarios. Even so, our method outperforms SCNN by **17.55%** and RESA (ResNet34) by **12.76%** on the SUNSET. Keypoints-based methods can be regarded as sparse counterparts of mask-based methods, and our method still surpasses GANet and CondLSTR across all datasets. The limitation of such methods in semi-structured scenarios lies in the ambiguity of road edge lines. In the feature space of these ambiguous local regions, the feature distances between adjacent pixels are extremely close, making it impossible to train an effective classifier for differentiation, which in turn renders pixel classification-based methods ineffective.

Metrics Comparison. We observe that $F1_E$ and $F1_R$ do not exhibit a strict quantitative relationship. Specifically, when more negative line instance samples are converted into positive road instance samples due to the averaging of IoU values, $F1_R$ will be greater than $F1_E$, and vice versa.

D. Ablation Study

Transformer Blocks. We investigated the impact of different numbers of Transformer blocks on the metrics. Table II demonstrates that a larger number of Transformer blocks exerts a positive effect on the improvement of model performance, yet such improvement has an upper limit. Specifically, both metrics on the GOOSE dataset exhibit a significant decline (-1.03%/-2.27%) when the number of blocks exceeds 3.

Loss Function. We conduct ablation experiments on the control point loss and the mask loss. As demonstrated in Table III, for the SUNSET dataset, the absence of the control point loss degrades the metric performance by 2.45% and 2.35% on $F1_E$ and $F1_R$ respectively compared with the baseline. On the YCOR dataset, this adverse effect is further amplified, with performance drops of 2.75% and 6.02%. This is because the road morphologies in semi-structured scenarios are more diverse, requiring higher-order curves to fit their edge lines. Without this loss term, the control points predicted by the model will be scattered randomly on both sides of the target curves, leading to a significant decline in model performance. Similarly, the removal of the mask auxiliary loss results in performance reductions of 1.63% and 2.59% on the SUNSET dataset, and even a 7.18% drop in $F1_R$ on the YCOR dataset. This indicates that the model's capability to distinguish road

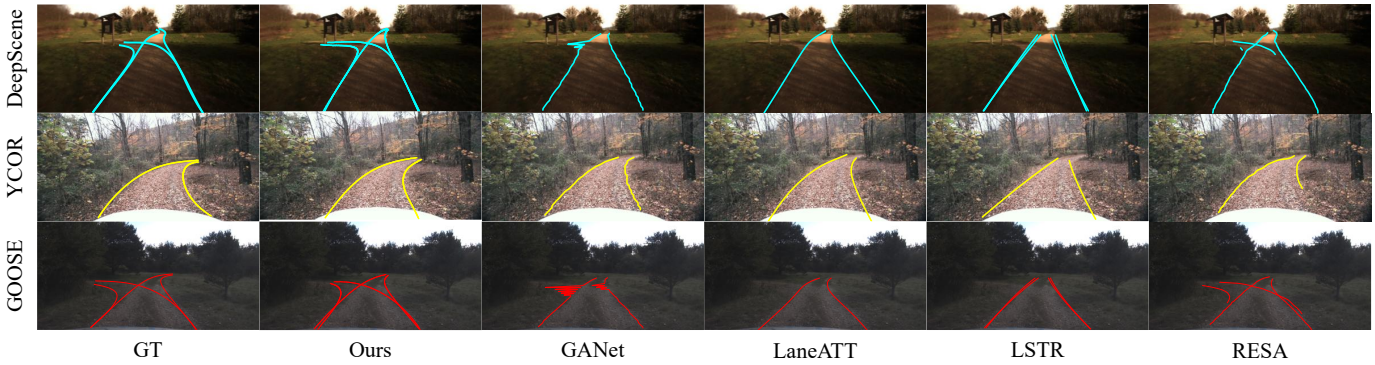

Fig. 3. Visualization results of TEAR and several representative lane detection methods on each sub-dataset of SUNSET. Specifically, in scenarios involving the detection of multiple roads, our method exhibits significantly superior visual performance compared to the other competing methods.

regions is substantially impaired, forcing it to directly regress the parameters of road edge lines without the guidance of road region masks.

Decoder Modules. The I-DI decoder enables TEAR to simultaneously focus on the road regions and the positions of road edge lines, thereby helping the model to accurately regress the control point parameters. We progressively ablated the instance converter $\mathcal{T}$ and the line-level multi-head cross-attention (MHA) mechanism of the I-DI decoder, which ultimately degraded into the vanilla Transformer decoder. As shown in Table IV, on the SUNSET, the line-level MHA mechanism improved the $F1_E$ and $F1_R$ by 2.07% and 1.29%, respectively. In contrast, the $\mathcal{T}$ only yielded improvements of 0.93% and 0.69%. We argue that the $\mathcal{T}$ merely enables the model to recognize the existence of line instances and represent them with queries, whereas the MHA mechanism provides more precise feature descriptions for line queries.

IV. Conclusions

In this paper, we propose a novel benchmark for road detection in semi-structured environment. We introduce the SUNSET dataset which conforms to the topological logic of roads. We develop an end-to-end road detection method that achieves competitive performance on the proposed benchmark. We propose the I-DI decoder, which enables the network to first learn road features and then model the relationship between road and edge lines on this foundation. Furthermore, we propose to use Hierarchical Bipartite Match to match both road instance and line instance with ground truth hierarchically. Extensive experiments demonstrate that our method outperforms competing methods on our proposed benchmark.

References

[1] Yanping Fu, Wenbin Liao et al., "Topologic: An interpretable pipeline for lane topology reasoning on driving scenes," *Advances in Neural Information Processing Systems*, vol. 37, pp. 61658–61676, 2024.

[2] Yanping Fu, Xinyuan Liu et al., "Topopoint: Enhance topology reasoning via endpoint detection in autonomous driving," *arXiv preprint arXiv:2505.17771*, 2025.

[3] Min Chen, Si Shubin et al., "Autonomous ground robots in unstructured environments: How far have we come?," *arXiv preprint arXiv:2410.07701*, 2024.

[4] Zhengyang Feng, Shaohua Guo et al., "Rethinking efficient lane detection via curve modeling," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 17062–17070.

[5] Abhinav Valada, Gabriel L Oliveira et al., "Deep multispectral semantic scene understanding of forested environments using multimodal fusion," in *International symposium on experimental robotics*. Springer, 2016, pp. 465–477.

[6] Daniel Maturana, Po-Wei Chou et al., "Real-time semantic mapping for autonomous off-road navigation," in *Field and Service Robotics: Results of the 11th International Conference*. Springer, 2017, pp. 335–350.

[7] Peter Mortimer, Raphael Hagmanns et al., "The goose dataset for perception in unstructured environments," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 14838–14844.

[8] Hang Xu, Xinyuan Liu et al., "Rethinking boundary discontinuity problem for oriented object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 17406–17415.

[9] Tu Zheng, Hao Fang et al., "Resa: Recurrent feature-shift aggregator for lane detection," in *Proceedings of the AAAI conference on artificial intelligence*, 2021, vol. 35, pp. 3547–3554.

[10] Qiankun Li, Xianwang Yu et al., "Pga-net: Polynomial global attention network with mean curvature loss for lane detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 1, pp. 417–429, 2023.

[11] Xingang Pan Jianping Shi Ping Luo Xiaogang Wang and Xiaoou Tang, "Spatial as deep: Spatial cnn for traffic scene understanding," in *AAAI Conference on Artificial Intelligence (AAAI)*, February 2018.

[12] Jinsheng Wang, Yinchao Ma et al., "A keypoint-based global association network for lane detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 1392–1401.

[13] Ziye Chen, Yu Liu et al., "Generating dynamic kernels via transformers for lane detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 6835–6844.

[14] Chao Chen, Jie Liu et al., "Sketch and refine: Towards fast and accurate lane detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024, vol. 38, pp. 1001–1009.

[15] Lucas Tabelini, Rodrigo Berriel et al., "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*, 2021, pp. 294–302.

[16] Ruijin Liu, Zejian Yuan et al., "End-to-end lane shape prediction with transformers," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2021, pp. 3694–3702.

[17] Peter Mortimer and Mirko Maehlisch, "Survey on datasets for perception in unstructured outdoor environments," *arXiv preprint arXiv:2404.18750*, 2024.

[18] Karsten Behrendt and Ryan Soussan, "Unsupervised labeled lane markers using maps," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[19] Tsung-Yi Lin, Michael Maire et al., "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[20] Xin He, Haiyun Guo et al., "Monocular lane detection based on deep learning: A survey," *arXiv preprint arXiv:2411.16316*, 2024.