



ElasticLaneNet: An Efficient Geometry-Flexible Lane Detection Framework

Yixin Feng¹, Yuan Lan¹, Luchan Zhang², Yang Xiang^{1,3*}

¹The Hong Kong University of Science and Technology, Hong Kong, China ²Shenzhen University, Shenzhen, China

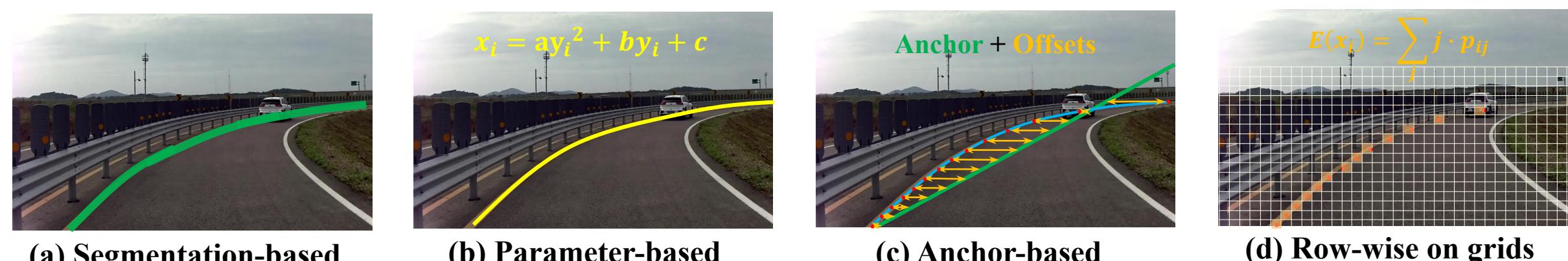
³HKUST Shenzhen-Hong Kong Collaborative Innovation Research Institute, China

Introduction

Background: Lane Detection requires recognizing the boundaries of the driving areas in real time. It is a critical technique of the visual perception in autonomous driving.

Challenges: Lane detection of **weak-feature lanes** and **diverse and complex lane geometries** are big challenges. The latter includes multiple-crossing lanes on the intersection, diverse-curvature curves, winding roads, Y-shape, L-shape and dense lanes, etc.

Current Lane Models:



Motivation: Previous methods mainly focused on those straight and parallel lanes. More robust lane detection strategies are needed in detecting challenging lane types.

Network Architecture and Objective Functions

ElasticLaneNet Architecture:

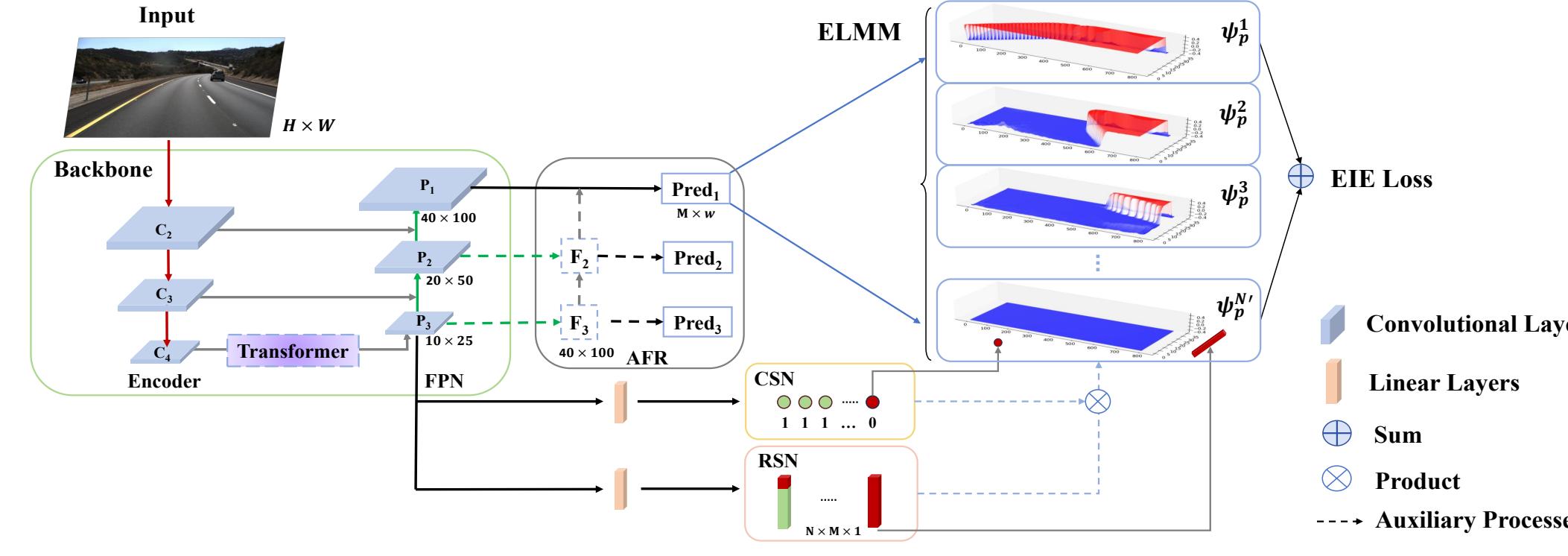


Figure 1. Network architecture of our ElasticLaneNet. CSN, RSN is designed to achieve end-to-end training strategy, AFR and Transformer block further strengthens the features integration.

Modules Description and Objective Functions:

- a. **Classification Sub-Network (CSN)** removes those redundant lane predictions: \mathcal{L}_{exist}
- b. **Range Sub-Network (RSN)** removes the out-of-range portion of lanes: \mathcal{L}_{range}
- c. **Elastic Lane Map Module (ELMM)** is the **ELM** prediction head, jointly trained with CSN and RSN modules (optional), from which the outputs are shaped by: \mathcal{L}_{eie}
- d. **Auxiliary Feature Refinement (AFR)** Integrates the features from different FPN layers: outputs are trained by auxiliary loss \mathcal{L}_{aux} or constructed by FPN layers fusion (FF);
- e. **Transformer bottleneck (TB)** applies self-attention mechanism in the bottleneck, thus ElasticLaneNet further explores the global features of the input images.
- ◆ **Total Loss Function** is consisted of the weighted sum of all the loss functions above:

$$\mathcal{L}_{total} = \lambda_{eie}\mathcal{L}_{eie} + \lambda_a\mathcal{L}_{aux} + \lambda_r\mathcal{L}_{range} + \lambda_e\mathcal{L}_{exist}.$$

Methodology

Elastic Lane Map Representation:

In our ElasticLaneNet, the lane is represented as zero-level contour lines of **ELM**, which is $\Psi = H(\phi) - 0.5$, and the lane curve is given by $\Psi = 0$. Here ϕ is the level set function [1] of the lane and $H(\cdot)$ is replaced to be the smoothed Heaviside function:

$$\phi(x, y) = \begin{cases} -d(x, y), & \text{if } (x, y) \in \gamma^- \\ 0, & \text{if } (x, y) \in \gamma \\ d(x, y), & \text{if } (x, y) \in \gamma^+ \end{cases} \quad H_\sigma(\phi) = \begin{cases} 0, & \text{if } \phi \leq -\sigma, \\ \frac{1}{2}(1 + \frac{\phi}{\sigma}), & \text{if } -\sigma < \phi < \sigma, \\ 1, & \text{if } \phi \geq \sigma \end{cases}$$

where γ^- and γ^+ respectively are the left and right sides of the lane curve γ , $d(x, y)$ is the x -distance to the lane curve with the same y -coordinate, and σ in $H_\sigma(\phi)$ is the hyperparameter.

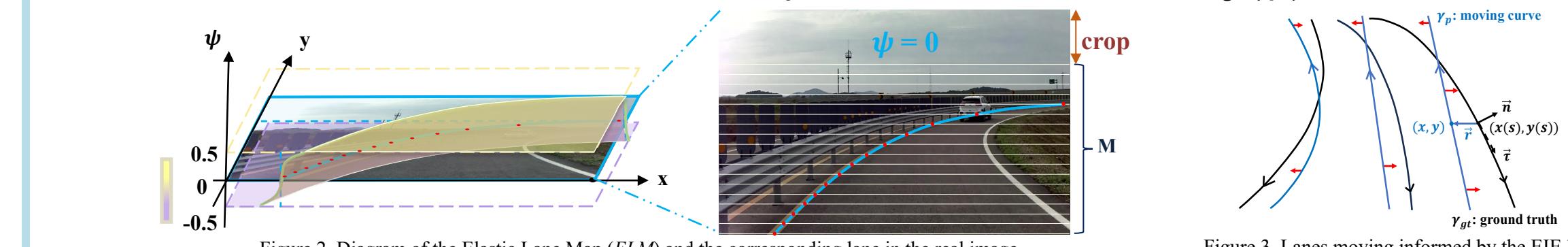


Figure 2. Diagram of the Elastic Lane Map (ELM) and the corresponding lane in the real image



Figure 3. Lanes moving informed by the EIE loss

Elastic Interaction Energy (EIE) Loss Function:

The EIE loss is inspired by the line defect phenomenon in physics. Analogous to long thin object, zero contour of the **ELM** is modeled as the moving curve that has elastic energy with the GT:

$$E = \frac{1}{8\pi} \int_{\gamma_{gt} \cup \gamma_p} \int_{\gamma'_{gt} \cup \gamma'_p} \frac{dl \cdot dl'}{r} = \frac{1}{8\pi} \int_{\gamma_{gt}} \int_{\gamma'_{gt}} \frac{dl_{gt} \cdot dl'_{gt}}{r} + \frac{1}{8\pi} \int_{\gamma_p} \int_{\gamma'_p} \frac{dl_p \cdot dl'_p}{r} + \frac{1}{4\pi} \int_{\gamma_{gt}} \int_{\gamma_p} \frac{dl_{gt} \cdot dl_p}{r}$$

In our ElasticLaneNet, through **ELM** representation, i.e. $\Psi_p = H_\sigma(\phi_p) - 0.5$ and $G_t = H_\sigma(\phi_{gt}) - 0.5$, we use the following elastic interaction energy (EIE) Loss:

$$\mathcal{L}_{eie}(\Psi_p, G_t) = \frac{1}{8\pi} \int_{\mathbb{R}^2} dx dy \int_{\mathbb{R}^2} \frac{\nabla(G_t - \alpha\Psi_p)(x, y) \cdot \nabla(G_t - \alpha\Psi_p)(x', y')}{r} dx dy.$$

The velocity used in the back-propagation is the gradient of \mathcal{L}_{eie} :

$$v(x, y) = -\frac{1}{4\pi} \int_{\mathbb{R}^2} \frac{r \cdot \nabla(G_t - \alpha\Psi_p)(x, y)}{r^3} dx dy,$$

Efficient Computation:

The gradient descent corresponding to the EIE loss in neural network is computed in the Fourier Space through FFT, thus the computational complexity reduced from $O(N^2)$ to $O(N \log N)$:

$$\frac{\partial \mathcal{L}_{eie}}{\partial \Psi_p} = \mathcal{F}^{-1} \left(\frac{\sqrt{m^2 + n^2}}{2} d_{mn} \right),$$

where d_{mn} is the Fourier transform of $G_t - \alpha\Psi_p$, and \mathcal{F}^{-1} is the inverse Fourier transform.

Point-wise ElasticLaneNet^{pw}:

When the output is not **ELM** but discrete lane points, the computational cost will decrease. The explicit point-wise variant version of our approach is named ElasticLaneNet^{pw}.

The variant EIE loss and the velocity here are:

$$E(x, y) = \frac{1}{8\pi} \int_{\mathbb{R}^2} \delta(\gamma) dx dy \int_{\mathbb{R}^2} \frac{\tau \cdot \tau'}{r} \delta(\gamma') dx dy'$$

$$\gamma_p(x, y) = -\left(\frac{1}{4\pi} \int_{\mathbb{R}^2} \frac{r}{r^3} \cdot (\mathbf{n}_{\gamma'_{gt}} \delta(\gamma_{gt}) - \alpha \mathbf{n}_{\gamma_p} \delta(\gamma_p)) dx dy' \right) \mathbf{n}_{\gamma_p}$$

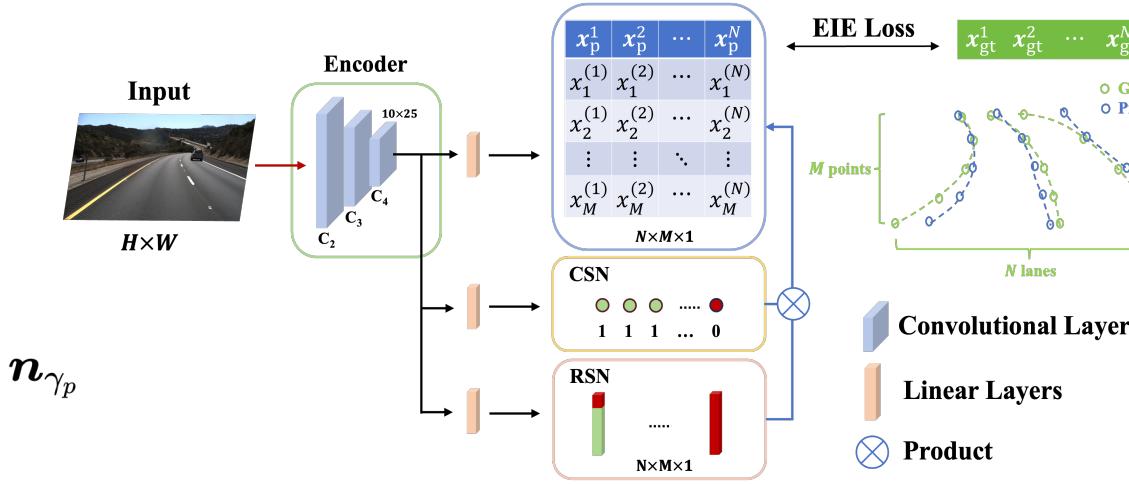


Figure 4. Network Architecture of our ElasticLaneNet^{pw}

Experimental Results

Results on SDLane Benchmark:

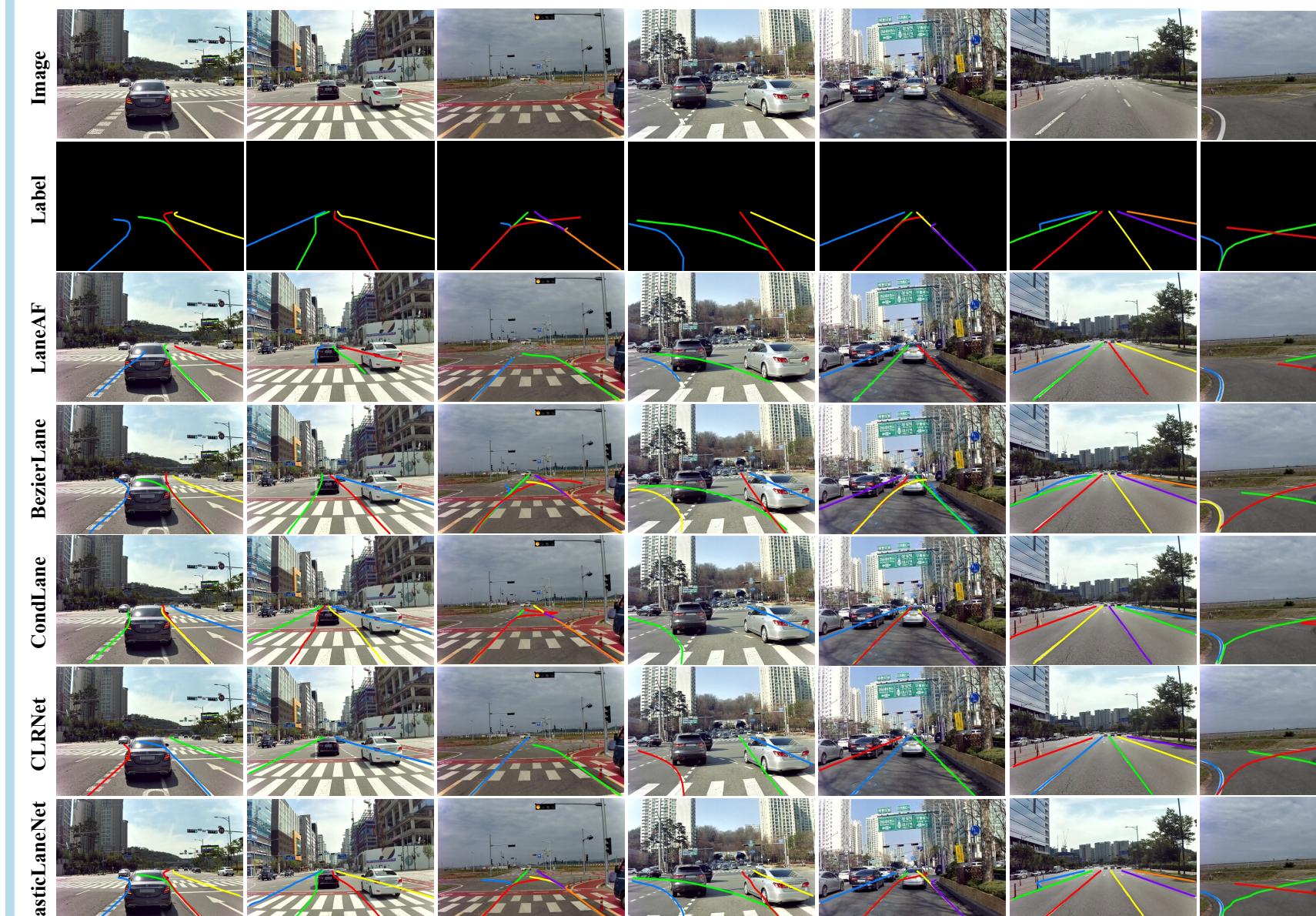


Figure 5. Experiment results comparison between our ElasticLaneNet and other current methods on SDLane.



Figure 7. Comparison between two version of ElasticLaneNet, the explicit lane point and the implicit ELM outputs, respectively.



Figure 8. Comparison between ELM trained via Mean Square Error (MSE) loss and EIE loss.

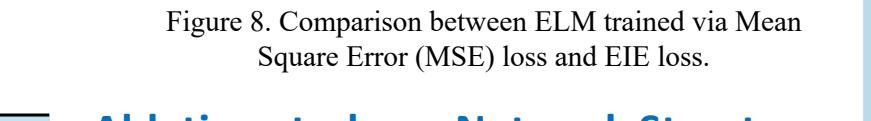


Figure 9. Ablation study on network structure. The order of different versions (Vi, i=1,2,3,4) follows the paper.

Other results on Tusimple & CULane:

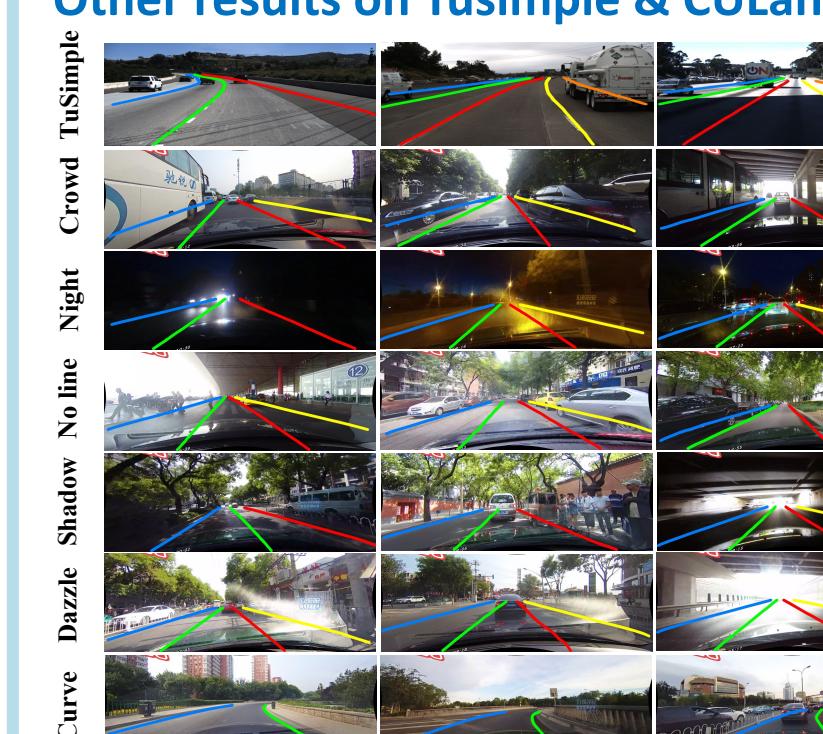


Figure 6. Some results of the challenging cases in TuSimple and CULane.

Method	F1	Precision	Recall	Type
RESA [†]	77.09	82.35	72.46	Seg.-based
LaneAF	74.17	87.15	64.55	Para.-based
BazierLane	77.13	77.84	76.43	
LaneATT [†]	73.49	85.78	64.28	
EigenLanes [†]	80.47	86.04	75.58	Anchor-based
CLRNet	85.04	94.16	77.53	
CondLane [†]	75.97	87.59	67.08	Row-wise
CondLane ^R	87.02	87.73	86.32	
UFLDV2	65.41	80.21	55.21	
ElasticLaneNet	89.51	91.61	87.50	Ours
ElasticLaneNet ^T	90.88	92.87	88.97	

Table 1. Results comparison on SDLane (with ‘†’ are from [2], and the others are our re-implementation).

Method	LaneAF	CLRNet	CondLane	ElasticLane	ElasticLane ^T	BezierLane
Type	Seg.	Anchor.	Row.	Ours	Para.	
FPS	11.29	59.26	61.95	75.42	66.62	109.37

Table 2. Frames Per Second (FPS) on SDLane (RTX 4000 server). The higher is the better.

Conclusions:

- We propose a novel lane representation **ELM** that can flexibly capture the exact lane geometry (shape and structure) and preserve the lane connectivity.
- A physics-inspired EIE loss is introduced to lane detection. It gives good guidance on training the **ELM**, overcomes the problem of weak features, ensures lane smoothness and enhances training stability.
- Our ElasticLaneNet excels at detecting lanes with complex geometry structures, with effectiveness and efficiency simultaneously. We achieve state of the art performance on structure diverse dataset SDLane.
- Ablation study illustrates the effects of **ELM** lane representations, EIE loss and different network modules.

References

- [1] Osher *et al.* Level set methods and dynamic implicit surfaces[J]. Appl. Mech. Rev., 2004, 57(3): B15-B15.
- [2] Jin *et al.* Eigenlanes: Data-driven lane descriptors for structurally diverse lanes. In CVPR 2022.