*International Journal of Basic and Applied Sciences*, 14 (4) (2025) 549-561

International Journal of Basic and Applied Sciences

*Website: www.sciencepubco.com/index.php/IJBAS*
*https://doi.org/10.14419/wb7z2179*
***Research paper***

# A Review of Deep Learning-Based Lane Detection Methods in Complex Environments

**Shiling Huang [1,2], Nur Ariffin Mohd Zin [2] *, Mohd Hamdi Irwan Hamzah [2]**

*[1] Intelligent Manufacturing College, Nanning University, Nanning, 530200, China*
*[2] University Tun Hussein Onn Malaysia, Parit Raja, Johor, 86400, Malaysia*
**Corresponding author E-mail: huangshiling@unn.edu.cn*

*Received: June 27, 2025, Accepted: August 12, 2025, Published: August 20, 2025*

**Abstract**

Lane detection is pivotal for enhancing the safety and functionality of Advanced Driver Assistance Systems (ADAS) and autonomous driving. Traditional image processing methods, while efficient, struggle in complex environments characterized by occlusions, lighting variations, and road clutter. Deep learning, particularly Convolutional Neural Networks (CNNs), has revolutionized lane detection by enabling automatic feature extraction from raw data, yet challenges persist in handling environmental variability and feature sparsity. This paper comprehensively reviews lane detection methodologies, encompassing both traditional techniques (e.g., Hough transforms, edge detection) and modern deep learning approaches. It emphasizes the critical role of integrating global and local contextual information to improve accuracy in challenging scenarios. Deep learning methods are categorized into three paradigms based on lane representation: segmentation-based, point-based, and parametric-based models. The review further explores how temporal feature fusion (leveraging consecutive video frames) mitigates occlusions and missing features, while spatial feature fusion captures long-range dependencies for holistic scene understanding. Key findings reveal that temporal-spatial fusion significantly enhances robustness, though real-time performance and adaptability to extreme conditions remain limitations. The paper concludes by identifying future research directions, prioritizing efficient architecture for real-time deployment and improved resilience in dynamic, unstructured environments.

***Keywords***: *Lane Detection; Complex Environments; Temporal Information Fusion; Global Context Integration*

## 1. Introduction

Lane detection is a foundational computer vision task critical for enhancing the safety and functionality of Advanced Driver Assistance Systems (ADAS) and autonomous driving [1]. It enables vehicles to identify lane boundaries, prevent collisions, and support precise navigation [2], [3]. Over the years, methodologies have evolved from traditional techniques (e.g., edge detection [4], Hough transforms [5]) to deep learning-based approaches. While traditional methods are efficient, they struggle in dynamic environments due to occlusions, lighting variations, and irregular lane markings. Deep learning, especially CNNs, has improved robustness through automatic feature learning, yet challenges persist in complex scenarios.

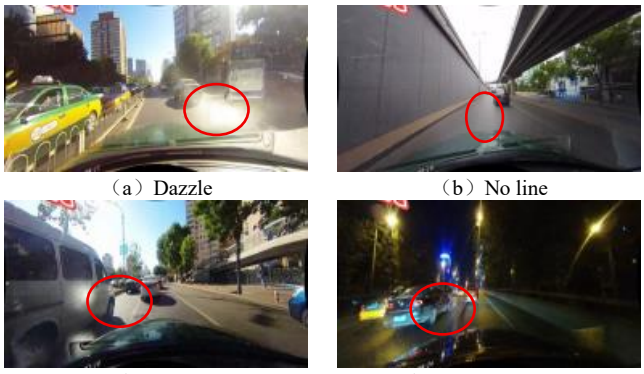
However, achieving robust lane detection in complex environments remains a significant challenge [6]. Complex environments refer to driving scenarios that introduce substantial challenges to lane detection systems, extending beyond structured road settings. These environments are characterized by multifaceted disturbances that degrade the visibility, continuity, or discriminability of lane markings. As illustrated in Fig. 1, four archetypal complex scenarios include:

1) Dazzle (Fig. 1a): Intense light sources (e.g., direct sunlight, headlight glare) cause overexposure or reflections, obscuring lane markings.
2) No Line (Fig. 1b): Absent, faded, or irregular lane markings due to road wear, construction, or non-standard geometries.
3) Crowded (Fig. 1c): Dense traffic or roadside objects (e.g., vehicles, barriers) partially or fully occlude lane boundaries.
4) Night and Crowded (Fig. 1d): The combination of low-light conditions and high levels of congestion significantly exacerbates the loss of visibility and intensifies occlusion effects.

This paper aims to: (1) systematically review traditional image processing and deep learning-based lane detection methods, highlighting their capabilities and limitations; (2) analyze critical challenges in complex driving environments (e.g., occlusions, lighting variations, and road clutter) that compromise detection robustness; and (3) evaluate the role of temporal and spatial feature fusion techniques in enhancing model resilience, ultimately identifying gaps and future research directions for real-time, reliable lane detection systems.

Following this introduction, Section 2 identifies and discusses critical challenges in lane detection under complex environments. Section 3 presents a comprehensive literature review, covering traditional image-processing methods (e.g., edge detection, Hough transform, color thresholding) and their limitations, followed by deep learning approaches categorized into segmentation-based, point-based, and parameter-based models, and concluding with enhancements through temporal and spatial context integration. Section 4 conducts a comparative

analysis of these methods, supported by comparison tables to highlight accuracy trends, robustness, and runtime performance. Section 5 summarizes gaps from the discussion and proposes future research directions. Section 6 concludes by synthesizing key findings.



（a）Dazzle （b）No line

（c）Crowded （d）Night add Crowded

**Fig. 1:** Illustrations of hard cases for lane detection. (a) Dazzle—where bright light sources cause glare, making it difficult to detect lane markings; (b) No line—where lane markings are absent or worn out, causing a lack of reference for detection; (c) Crowded—where dense traffic or surrounding objects obscure lane boundaries; and (d) Night and Crowded—a combined challenge where low-light conditions and congestion together complicate the detection process.

## 2. Problem statement

Lane detection remains critically challenging in complex driving environments due to dynamic and interdependent factors that degrade algorithmic robustness [1], [6]. Despite advancements in computer vision and deep learning, the following unresolved challenges persistently impair detection reliability and safety:

1) Degraded Feature Discriminability: Visual interference, such as glare and shadows, reduces the contrast between lane markings and the road surface, while adverse weather conditions like rain or snow introduce significant noise and reflections [7] causing traditional edge detectors (e.g., Canny, Sobel) and color-based thresholding methods [8], [9] to fail in extracting stable lane features, resulting in false negatives or fragmented detections.
2) Structural Ambiguity: The absence or severe wear of lane lines (No Line scenarios) breaks the continuity of lane geometry, and curved or non-standard lanes deviate significantly from common parametric models (e.g., straight-line Hough transforms) [10], [11], leading to inaccurate fits or complete misses by model-based methods (e.g., RANSAC [12], polynomial fitting [13]) due to insufficient reliable feature points.
3) Dynamic Occlusions: Moving obstacles, such as vehicles in crowded scenes, persistently block lane segments, and static objects like debris or barriers create partial occlusions, which impede segmentation-based deep learning methods (e.g., LaneNet [14], SCNN [15]) from recovering occluded lanes effectively without leveraging temporal context.
4) Low Signal-to-Noise Ratio (SNR): Night scenes and poor weather drastically reduce illumination intensity, while environmental clutter such as tire marks or pavement cracks mimics genuine lane features, resulting in spurious edges dominating the feature extraction process and increasing false positives [16]; consequently, deep learning methods require extensive data augmentation to generalize to these challenging low-SNR scenarios [1], [17].
5) Multimodal Interference: The combination of challenges (e.g., Night and Crowded) simultaneously introduces multiple stressors like low light and occlusion, overwhelming single-frame detection paradigms; therefore, models lacking robust spatiotemporal fusion mechanisms fail to leverage contextual cues from consecutive frames, often leading to catastrophic detection failures [18], [19].

Complex environments disrupt lane detection by weakening feature reliability, breaking structural priors, and introducing adversarial noise. While deep learning methods mitigate some issues through hierarchical feature learning, they remain vulnerable to extreme multimodality and require explicit architectural innovations (e.g., temporal-spatial fusion [20]) to achieve robustness.

## 3. Literature review

Lane detection has evolved from traditional image-processing techniques to deep learning-based paradigms, driven by the need for robust performance in complex driving environments [1], [11]. Seminal techniques, such as the Hough Transform [5] and curvilinear structure detectors like Steger's method [21], laid the groundwork for classical lane detection by enabling robust extraction of lines and curves from road imagery. These methods exemplify the early generation of lane detection approaches, which typically relied on handcrafted feature extraction and geometric modeling. However, traditional pipelines often struggled under dynamic conditions, including illumination changes, occlusions, and road clutter, which limited their robustness and adaptability in real-world scenarios. The advent of deep learning revolutionized the field by enabling end-to-end feature learning, significantly improving adaptability to challenging scenarios [6], [9]. This section reviews key methodologies across two eras: image-processing-based approaches (Section A) and deep learning methods categorized into segmentation-based, point-based, and parameter-based paradigms (Section B). By contrasting their principles, strengths, and limitations, we establish a foundation for understanding current advancements and unresolved challenges in lane detection.

### 3.1 Image-processing-based lane detection methods

Traditional image processing techniques have formed the foundation of lane detection algorithms due to their computational efficiency and interpretability. These methods rely on sequential steps involving image enhancement, feature extraction, and model fitting to identify lane boundaries [9], [22]. This section reviews key methodologies in traditional lane detection, focusing on their implementation, advantages, and limitations.

### 3.1.1 Overview of image processing-based lane detection methods

Traditional lane detection methodologies employ sequential image processing modules to identify lane boundaries through systematic feature extraction and geometric modeling. The pipeline typically comprises four stages: image preprocessing, edge detection, feature clustering, and model fitting [12]. In the preprocessing stage, raw images are transformed to enhance lane visibility. Grayscale conversion [23] simplifies computational complexity by reducing color channels. Adaptive histogram equalization (AHE) [8] improves contrast in low-light or shadowed regions, ensuring consistent feature extraction under varying illumination. Region of Interest (ROI) [24] selection further optimizes processing efficiency by focusing computational resources on the lower portion of the image, where lane markings are most prominent. Edge detection [4] serves as the cornerstone for isolating lane boundaries. The Canny operator [16] It is widely adopted due to its dual-threshold mechanism, which suppresses noise while preserving fine edges. In contrast, the Sobel [25] The operator emphasizes gradient-based edge detection through horizontal and vertical kernel convolutions. For curved or discontinuous lane markings, directional Sobel filters (e.g., 45° and 135°) are implemented to enhance edge continuity. Subsequent feature clustering employs statistical methods to group edge pixels into coherent lane candidates. The Random Sample Consensus (RANSAC) algorithm [26] iteratively selects subsets of edge points to estimate lane geometry while rejecting outliers. Hough Transform [5], while effective for straight-line detection, maps edge pixels to a parameter space (e.g., ρ-θ coordinates) to identify linear lane boundaries. However, its limitations in modeling curved lanes necessitate supplementary techniques, such as B-spline curves [27] or polynomial fitting [28], to approximate nonlinear geometries.

### 3.1.2 Challenges and Limitations of Image Processing-Based Methods

Despite their computational simplicity, traditional methods exhibit critical limitations in dynamic and complex environments. Adaptive thresholding partially mitigates lighting variations but requires empirical parameter tuning for diverse scenarios [9]. Shadows, occlusions, and weather-induced artifacts (e.g., rain, snow) degrade edge detection accuracy by introducing spurious edges or suppressing valid lane features. For instance, wet road surfaces generate reflections that mimic lane markings, leading to false positives. While Hough Transform excels in detecting straight lanes, its reliance on parametric line models renders it ineffective for curved or discontinuous lane geometries [11]. Polynomial approximations (e.g., quadratic or cubic curves) and B-splines address this limitation but increase computational complexity and sensitivity to noise [13]. Iterative algorithms like RANSAC [26] and Hough Transform [5] impose significant computational overhead due to their exhaustive search mechanisms. For example, RANSAC's iterative hypothesis-and-validation loop scales poorly with dataset size, hindering real-time performance in high-speed applications [29]. Hardware acceleration via GPUs or FPGAs offers partial mitigation but raises implementation costs, limiting feasibility in resource-constrained systems.

## 3.2 Deep-learning methods

With the rapid development of deep learning, breakthroughs have been made in the field of computer vision, which greatly promotes the development of lane detection technology. Traditional lane detection methods mostly rely on manual feature extraction and simple image processing algorithms, which often perform poorly in complex environments, illumination changes, and occlusion [3], [6], [28]. The introduction of CNNs makes automatic feature learning possible, and the network can extract multi-level and multi-scale effective features from a large amount of data, which significantly improves the accuracy and robustness of lane detection [14]. Furthermore, deep learning not only enhances the model's global semantic understanding of lanes but also improves the ability to capture local details, making lane positioning more accurate in complex road scenes.

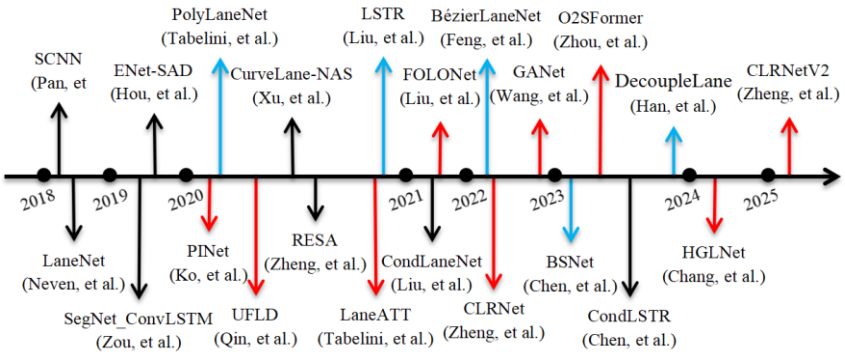



**Fig. 2:** Major methods of deep learning-based network models, including segmentation (in black), Point-based (in red), and curve fitting (in blue)

### 3.2.1 Overview of network model classification

According to the lane line representation methods output by deep learning network models, lane detection methods based on deep learning are usually divided into three categories [30]. The first category is the method based on semantic segmentation, which considers lane detection as a pixel-level classification task and predicts whether each pixel in the image belongs to the lane line. Typical representatives include SCNN [15] and RESA [18]. The second category is the method based on point detection. This kind of method quickly locates the existence of lane lines by setting several anchor points or key points on the image, and then regresses and classifies them, such as CLRNet [31]. The third category is the method based on parameter regression, which realizes the detection by fitting the mathematical model parameters (such as polynomial coefficients) of the lane lines. Such methods have advantages in inference speed and model simplicity, such as PolyLaneNet [28] and LSTR [19]. Each type of method has its characteristics, providing diverse solutions for different application scenarios and requirements, and promoting lane detection in the direction of higher accuracy and real-time performance. The evolution of lane detection methodologies has witnessed a paradigm shift from CNNs to the recent integration of Transformer architectures [32], reflecting continuous efforts to balance detection accuracy and computational efficiency. The evolution of deep learning-based lane detection models, as illustrated in Fig. 2, reveals a clear progression from early segmentation frameworks to hybrid architectures incorporating curve fitting and attention mechanisms. This figure categorizes methods into three color-coded paradigms: segmentation-based (black), point-based (red), and curve fitting (blue), with time-sequenced annotations highlighting their development from 2018 to 2025.

### 3.2.2 Segmentation-based methods

Semantic segmentation is a computer vision technique aimed at classifying each pixel in an image into a specific category [33]. Unlike image classification, which assigns a single label to the entire image, semantic segmentation requires assigning a category label to every pixel in the image. This means that semantic segmentation not only identifies what is in the image but also knows the exact location of each object within the image. LaneNet [14] is one of the pioneering works in this area, employing a dual-branch network architecture that consists of a semantic segmentation branch and a lane embedding branch to effectively separate and identify lane markings, as shown in Fig. 3. This design enables accurate multi-lane detection by clustering lanes based on their embedding vectors. The model has demonstrated its capacity to manage complex scenes and handle occlusions and intersecting lanes, providing a robust solution for structured environments. However, its performance degrades under extreme lighting and weather conditions, which still pose challenges for real-world deployment. ENet-SAD [17], an extension of the Enet [34] The architecture incorporates a spatial attention mechanism to improve lane detection accuracy. This mechanism enables the model to focus more effectively on regions of interest, such as lane markings, thereby enhancing performance in real-time applications. The efficiency of ENet-SAD [17] makes it suitable for deployment in time-sensitive environments, as it achieves high computational efficiency. Despite these advantages, the model's ability to distinguish lanes from complex backgrounds, such as when occlusion or lighting variations occur, is limited. CurveLane-NAS [35], which leverages Neural Architecture Search (NAS), introduces an automatic architecture search process to optimize the network for lane detection tasks. This approach enables the generation of a network structure tailored to the unique geometric characteristics of lanes, especially in cases involving curved or intersecting lanes. By automating the architecture search, CurveLane-NAS [35] improves the model's adaptability and accuracy. However, the NAS process is computationally intensive, requiring significant resources and time for training, thus increasing the overall cost of model development. LaneAF [36] employs an attention-based fusion mechanism that integrates both global and local attention features to enhance lane detection accuracy, particularly in complex and cluttered environments. This mechanism improves the model's robustness by focusing on critical features while suppressing irrelevant background information. LaneAF's superior performance in challenging traffic scenarios highlights its potential, but its high computational demands, particularly when processing high-resolution images, can be a limiting factor for real-time applications. CondLSTR [37], based on the Transformer architecture, proposes a novel approach for lane detection by leveraging conditional sequence generation to predict lane coordinates over both spatial and temporal dimensions. This model effectively exploits global contextual information, making it particularly robust in dynamic environments. By utilizing long-range dependencies, CondLSTR [37] is capable of handling complex dynamic scenes. However, the computational overhead of processing multi-frame sequences remains a challenge, especially when real-time performance is a critical requirement.

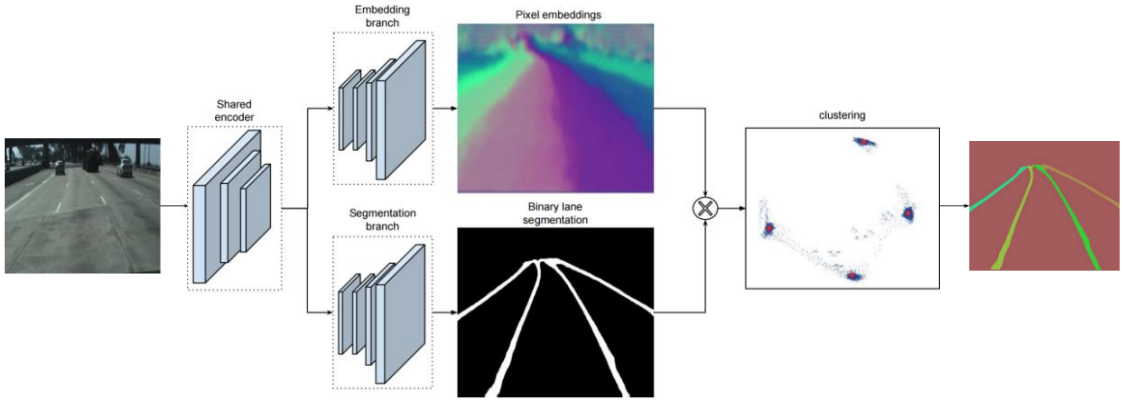



**Fig. 3:** LaneNet network structure [14]

### 3.2.3 Point-based methods

Point-based methods for lane detection focus on detecting key points along the lane rather than segmenting the entire lane region. These methods treat the lane as a set of discrete points, which simplifies the detection process and enhances robustness against occlusions and irregular lane shapes. The key points are typically predicted through regression, and once detected, techniques like polynomial regression or B-splines are used to reconstruct the full lane. Inspired by the object detection model Faster R-CNN [38], Line-CNN [39] introduces a novel representation method for lane line anchors and addresses the challenges of lane line representation in lane detection. The Line-CNN [39] network structure, as shown in Fig. 4, uses ResNet [40] as the backbone to extract feature maps from input images. Its core component, the Line Proposal Unit (LPU), is similar to the Region Proposal Network (RPN) in Faster R-CNN [38]. The LPU generates a set of line proposals at each pixel on the left, right, and bottom of the feature map. Each line proposal is represented by a vector, including the confidence that it is a lane line, its length, and its offset in the x-axis direction. To represent lane lines, Line-CNN [39] sets a series of evenly distributed horizontal cut lines along the y-axis of the original image, dividing the image into multiple horizontal regions. The intersections of the actual lane lines with these cut lines are used to describe the shape of the lane lines. By calculating these intersections, a set of discrete points on the x-axis is obtained, effectively outlining the lane lines. This method simplifies the representation of lane lines and improves detection accuracy and efficiency. LaneATT [41] also treats rays in the image as anchor points, classifying and locating lane lines from dense line anchors, and combining local and global features using an attention mechanism to address scenarios with occlusions or no visible lane lines. However, these line anchor-based methods not only require post-processing steps like NMS but also rely on dataset statistics to design anchor representations, where fixed-shaped anchors may not be flexible enough to describe lane lines with high degrees of freedom. UFLD [42] treats lane line detection as a row selection classification problem based on global image features, incorporating structural losses to enhance the similarity and shape constraints of network outputs, achieving a lightweight lane line detection model. Despite the faster speed of row anchor-based methods, their performance on challenging large datasets still lags other categories of methods. CLRNet [31] represents lanes as a sequence of equidistant 2D points, sampled along the vertical axis of the image. This representation aligns with the continuous nature of lane markings and allows the model to directly predict the lane shape through regression. This approach ensures smooth and coherent lane detection, overcoming issues like discontinuities or jumps common in pixel-level methods, while enhancing overall accuracy and robustness.

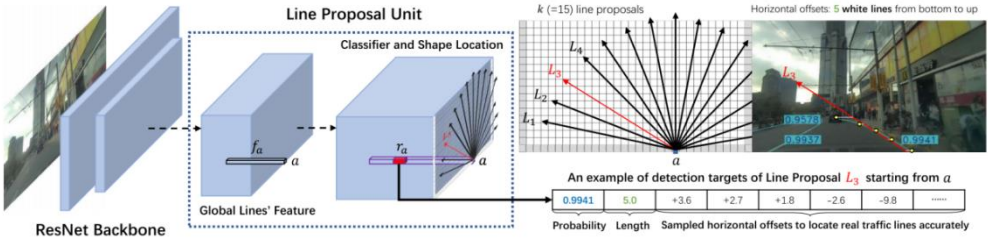



**Fig. 4:** Overall scheme of Line-CNN [39]

### 3.2.4 Parameter-based methods

Parameter-based methods represent lane lines using curves and model their shape by employing neural networks to directly predict the parameters of these curves, as shown by the yellow line (polynomial curve) and the red line (third-order Bézier curve) on the right side of Fig. 5. PolyLaneNet [28] predicts lane line coefficients through deep polynomial regression, estimating vertical offsets and confidence scores for each lane line. LSTR [19] utilizes a lane line shape model that regresses curve parameters by integrating road structure and camera pose, thereby enhancing the model's understanding of lane shapes. While parameter-based methods are fast, they exhibit lower performance. Due to the challenges in optimizing and abstracting polynomial coefficients, these methods suffer from slow convergence speeds and high structural latency. To address the optimization difficulties of polynomial curve methods, BézierLaneNet [30] models the geometric shape of lane lines using a parameterized Bézier curve and introduces a feature flipping fusion module based on deformation convolution, leveraging the symmetry of lane lines in the forward view. BSNet [27] introduces a lane detection model based on B-spline curves, with a focus on balancing local and global lane representation. The model employs B-spline curves, defined by 8 control points, where the local influence is determined by neighboring control points (k+1), while the global structure is adaptable by increasing the number of control points to handle complex lane topologies. Although parameter-based methods have relatively quick inference times, they are also sensitive to the predicted parameters, e.g., the error in predicting high-order coefficients can cause significant changes in the shape of the lanes, making it challenging to achieve higher performance.

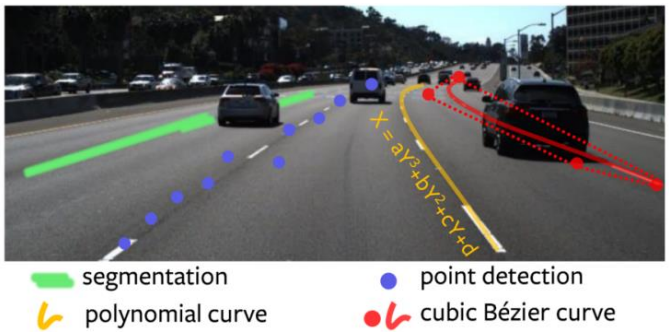



**Fig. 5:** Different expressions of lane lines [30]

## 3.3 Temporal and Spatial Context Enhancements

Research has shown that both global and local information are essential for improving detection accuracy [15], [31], [43], [44]. Lane detection in complex environments, such as low-light conditions, strong occlusions, and dynamic road scenarios, faces challenges like feature sparsity, topology breakage, and temporal jitter. Deep learning has significantly enhanced robustness by enabling temporal-spatial joint modeling, which allows models to address these challenges more effectively. Current research primarily focuses on two key approaches: temporal information fusion and spatial global context modeling. Meanwhile, with the rapid development of Transformer architectures in the computer vision domain, their core component—self-attention [32]—has demonstrated remarkable capabilities in modeling long-range dependencies and capturing both spatial and temporal contextual information. These characteristics make Transformer-based methods particularly well-suited for complex lane detection tasks, especially under conditions requiring global context understanding and multi-frame feature integration.

### 3.3.1 Methods of Integrating Temporal Feature Information

Most research on lane detection currently focuses on using a single image for lane line detection. However, in practical applications, lane detection systems process sequences of video frames with temporal continuity [20]. By integrating information from adjacent frames, the limitations of single-frame images, such as occluded lane parts that may be visible in past frames, can be mitigated. This temporal feature fusion improves the accuracy and stability of lane detection by enhancing the lane line and its related features in the current frame. The principle behind temporal fusion is to leverage the temporal context to address missing or unclear lane features, thereby helping the model better understand lane continuity and resolve issues such as occlusions, topology breakage, and temporal jitter. Common methods for temporal feature integration include concatenation, attention mechanisms, and motion estimation, which enable the model to use past frame information for enhanced detection performance. Fig. 6 illustrates the overall architecture of the proposed TGC-Net [45], a video-based lane detection network designed to leverage spatiotemporal context for improved robustness and accuracy. The network follows an encoder-decoder paradigm, where the encoder is built upon ResNet-50 for extracting rich spatial features from individual video frames, and the decoder adopts the bilateral up-sampling structure introduced in [18] to generate dense lane predictions.

The core innovation of TGC-Net [45] lies in the Temporal Recursive Feature-Shift Aggregation Module (T-RESA), which enables effective modeling of both temporal and spatial dependencies across consecutive frames. Given a fixed-length sequence of video frames (e.g., 3 frames as validated in the experiments), T-RESA recursively aggregates feature representations along three orthogonal directions: vertical, horizontal, and temporal. Specifically, for each time step, directional feature maps (denoted as Dt, Ut, Rt, Lt) are computed to propagate

information within the spatial domain, while temporal connections are established across frames to enhance lane continuity and suppress temporal jitter.

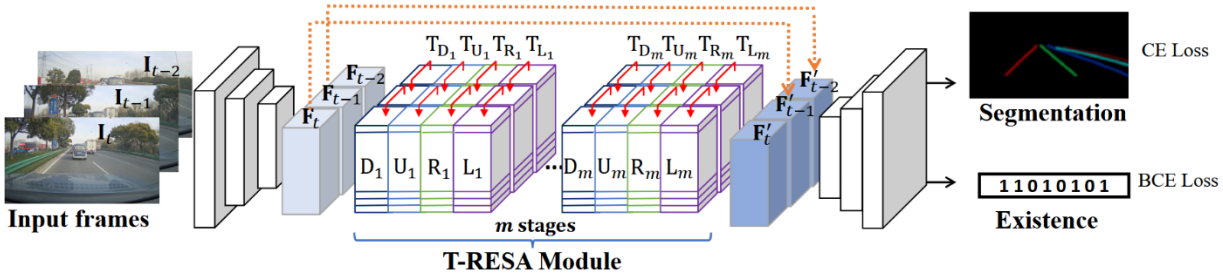



**Fig. 6:** TGC-Net structure [45]

Fig. 7 presents a detailed schematic of the upward-directional Temporal REcurrent Feature-Shift Aggregator (T-RESA) mechanism, which constitutes a key component of the TGC-Net [45] architecture. This module is designed to perform recursive spatiotemporal feature aggregation across multiple video frames, enabling robust lane structure modeling in complex driving environments. As shown in the upper left of the figure, the input is a sequence of multi-frame feature tensors with shape C×H×W, where C is the number of channels, H is the spatial height, and W is the width. These feature maps are spatially sliced along the vertical (H) dimension to obtain multiple C×1×W slices. Each slice represents one row of the original feature map, capturing full horizontal context at a specific vertical position. These slices serve as the basic units for directional feature propagation and are color-coded to indicate different spatial directions—Down (Dm), Up (Um), Right (Rm), and Left (Lm)—within the m-th stage of the T-RESA process.

The bottom portion of Fig. 7 highlights the recursive feature propagation process along the upward direction, spanning three consecutive time steps t−2, t−1, and t. For each time step, slices from different vertical levels (i) (denoted as Ft−2, i, Ft−1, i, and Ft, i) are progressively updated through a combination of intra-frame and inter-frame message passing. Vertical arrows indicate upward spatial aggregation within the same frame (from bottom to top), while diagonal arrows represent temporal connections across adjacent frames. The feature shift operation uses a stride of 1, enabling fine-grained feature alignment between neighboring spatial slices and across time. This mechanism allows the network to capture lane geometry evolution over time while preserving high-resolution spatial context. The upward T-RESA process shown here is one of four directional paths (up, down, left, right), all of which are executed independently in each T-RESA stage and subsequently fused to obtain comprehensive spatiotemporal representations. Multiple stages (m) of T-RESA are applied recursively to enable deep context modeling and iterative refinement of lane features.

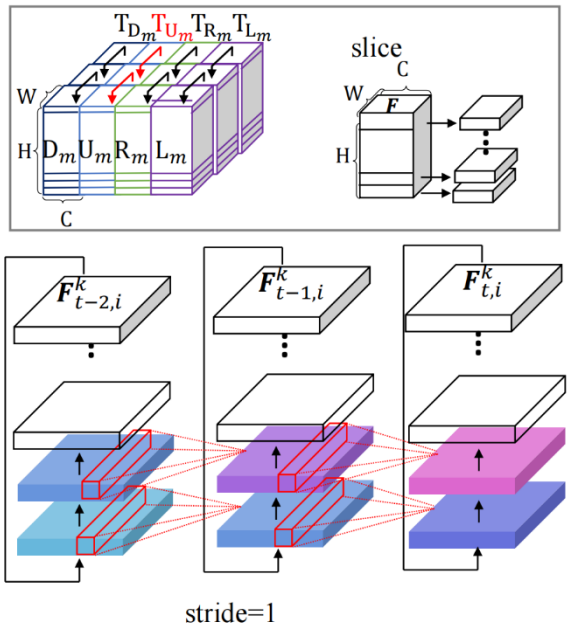



**Fig. 7:** Schematic illustration of upward Temporal REcurrent Feature-Shift Aggregator (T-RESA) when stide is 1 for *m-th* stage. The propagation for the other three directions is similar [45].

By jointly propagating features along both spatial and temporal axes, T-RESA facilitates the learning of temporally consistent and geometrically coherent lane representations, addressing challenges such as occlusion, curvature variance, and motion blur in video-based lane detection tasks.

### 3.3.2 Methods of Capturing Spatial Global Contextual Information

Although CNN-based methods have advanced scene understanding in driving to a new level with their powerful representation learning capabilities, they still struggle with detecting lane lines that have elongated structures, various shapes, and potential occlusions. Since lane line features are distributed over a wide range within the spatial information of images, recent research has emphasized the performance improvements that can be achieved by utilizing the long-range dependencies in these feature representations [20], [31], [46]. Pan, X. et al. (2018) proposed the Spatial Convolutional Neural Network (SCNN) [15] method, which utilizes long-range dependencies among pixels in

the feature map for lane prediction. As shown in Fig. 7, the SCNN [15] architecture uses CNNs to generate feature maps and performs slice convolutions in different directions within the feature map, such as vertical (SCNN_D, SCNN_U) and horizontal (SCNN_R, SCNN_L) directions. This allows information to be passed between pixels within the same layer, enabling the network to establish long-range dependencies across rows and columns of features, thereby enhancing the spatial relationships between pixels. The RESA [18] The model builds upon the SCNN [15] model by effectively aggregating spatial information by repeatedly shifting slice feature maps in vertical and horizontal directions, capturing lane shape priors and spatial relationships between pixels. The CLRNet [31] The model introduces a novel lane detection algorithm. By adopting the Feature Pyramid Networks (FPN) [47] fusion approach and ROIGather module to construct long-range dependencies between pixels from multi-scale feature maps.

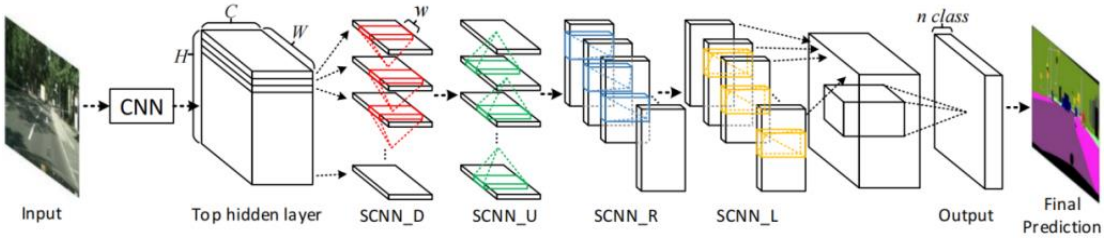



**Fig. 7:** The SCNN network structure [15]

### 3.3.3 Methods of Transformer-based

Transformer-based models offer a unified and flexible framework for end-to-end lane detection, exhibiting strong capability in capturing long-range dependencies, fusing multi-frame features, and achieving a balance between accuracy and computational efficiency. Unlike conventional CNNs that rely on local receptive fields, Transformers are well-suited for learning global representations and dynamic contexts in challenging driving scenarios. LSTR [19], as illustrated in Fig. 8, represents one of the earliest attempts to incorporate Transformer architectures into the lane detection domain.. It introduces a curve-based lane representation and formulates lane detection as a sequence prediction task. The LSTR [19] The framework includes three key components: a CNN backbone for feature extraction, a Transformer encoder-decoder for global modeling, and a curve prediction head. This architecture allows LSTR [19] to learn high-level semantics and lane topology holistically. However, due to the sparsity and specificity of lane features, their global modeling may struggle to refine subtle lane details, which can lead to reduced accuracy in complex environments.

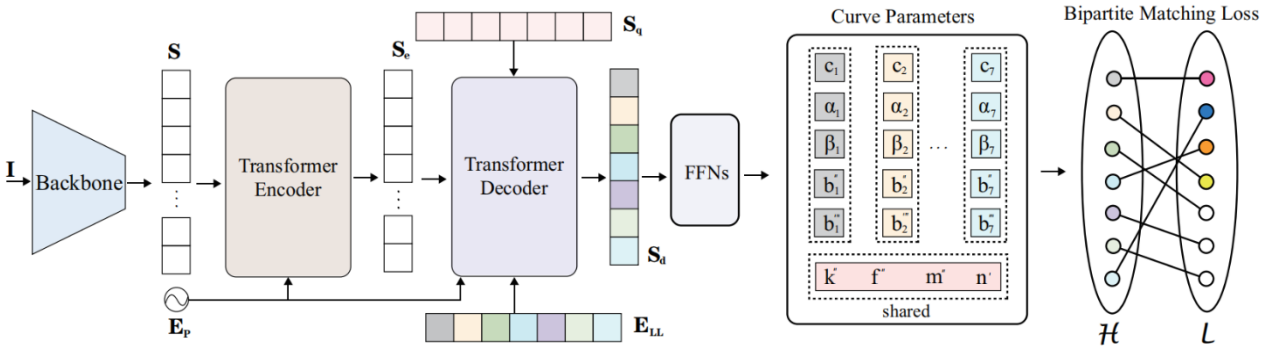



**Fig. 8:** The LSTR network structure [19]

LaneFormer [48] proposes a dual-axis attention mechanism, performing row-wise and column-wise self-attention operations to capture structured spatial dependencies. It also introduces ROI-aligned feature extraction to reduce computational cost. While effective at modeling structured context, it underperforms in fine-grained detail extraction due to its limited local feature refinement. CondLSTR [37] enhances modeling capacity by integrating conditional dynamic convolutions within a Transformer framework. It utilizes a high number of learnable queries (20–100) to boost detection performance. However, this design greatly increases computational complexity and memory consumption, which is less favorable for real-time applications. CondFormer [46] improves upon CondLSTR [37] by reducing the number of queries (fewer than 10) and combining conditional convolution with global attention, enabling efficient local-global feature fusion. This reduces the model's complexity while retaining competitive accuracy, making it more practical for embedded deployment.

## 3.4 Performance Comparison on Benchmarks

To strengthen the comparative analysis, we provide a quantitative synthesis of key performance metrics (e.g., F1 score and inference time) from representative deep learning-based lane detection methods. Table 1 and Table 2 summarize the reported results on two widely used public benchmarks: VIL-100 [49] and CULane [15], which cover various complex scenarios.

The VIL-100 [49] dataset is a pioneering resource for video lane detection. It is a scenario-specific dataset that contains 100 videos with 100 frames per video, totaling 10,000 frames. Among them, 97 videos were captured using a monocular forward-facing camera mounted near the rear-view mirror, while the remaining 3 videos were collected from the Internet under hazy conditions to enhance the dataset's diversity and realism. The dataset encompasses 10 typical driving scenarios, including normal, crowded, curved roads, damaged roads, shadows, road markings, dazzle light, haze, night, and crossroads. The training and testing splits follow an 8:2 ratio, ensuring that all scenarios are well represented in both subsets. To facilitate a more quantitative comparison among state-of-the-art methods on this dataset, Table 1 summarizes their performance across three key metrics: mean Intersection over Union (mIoU), F1 score at IoU threshold 0.5, and F1 score at threshold 0.8.

**Table 1:** Performance Comparison of Lane Detection Methods on VIL-100 [49] Dataset

| Method | mIoU ↑ | F1 0.5 ↑ | F1 0.8 ↑ |
|---|---|---|---|
| LaneNet [14] | 0.633 | 0.721 | 0.222 |
| ENet-SAD [17] | 0.616 | 0.755 | 0.205 |
| LSTR [19] | 0.573 | 0.703 | 0.131 |
| RESA [18] | 0.702 | 0.874 | 0.345 |
| LaneATT [41] | 0.664 | 0.823 | - |
| MFIALane [50] | - | 0.905 | 0.565 |
| MMA-Net [49] | 0.705 | 0.839 | 0.458 |
| TGC-Net [45] | 0.738 | 0.892 | 0.469 |
| RVLD [51] | 0.787 | 0.924 | 0.582 |
| LaneTCA [52] | 0.796 | 0.933 | 0.621 |

The CULane [15] The dataset is a large-scale and challenging benchmark designed for academic research on lane detection. It was collected in Beijing using cameras mounted on six different vehicles driven by different drivers, ensuring high data diversity. The dataset contains over 55 hours of driving videos, from which 133,235 frames were extracted. It is split into 88,880 training, 9,675 validation, and 34,680 test images. The test set is further divided into normal and eight challenging scenarios: crowded, dazzle light, shadow, no line, arrow, curve, cross, and night. To enable a more objective comparison of recent deep learning-based methods, Table 2 reports the F1 score (at IoU threshold = 0.5) across all test scenarios, along with inference speed (FPS). The FPS is measured under the PyTorch framework and reflects model efficiency in practical deployment. For the cross scenario, only false positives are reported due to its unique evaluation setting.

**Table 2:** Comparative Performance of Deep Learning-Based Lane Detection Methods

| Year | Method(backbone) | FPS | Total | Normal | Crowd | Dazzle | Shadow | No line | Arrow | Curve | Cross | Night |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Segmentation-based methods | | | | | | | | | | | | |
| 2018 | SCNN [15] (VGG16) | 7.5 | 71.60 | 90.60 | 69.70 | 58.50 | 66.90 | 43.40 | 84.10 | 64.40 | 1990 | 66.10 |
| 2021 | RESA [18] (ResNet-50) | 35.7 | 75.30 | 92.10 | 73.10 | 69.20 | 72.80 | 47.70 | 88.30 | 70.30 | 1503 | 69.90 |
| 2021 | LaneAF [36] (DLA-34) | 20 | 77.41 | 91.80 | 75.61 | 71.78 | 79.12 | 51.38 | 86.88 | 72.70 | 1360 | 73.03 |
| 2022 | MFIALane [49] (ResNet-34) | -- | 76.10 | 92.80 | 74.00 | 67.80 | 76.40 | 48.70 | 88.60 | 71.20 | 1790 | 72.00 |
| Point-based methods | | | | | | | | | | | | |
| 2020 | LaneATT [41] (ResNet-122) | 20 | 77.02 | 91.74 | 76.16 | 69.47 | 76.31 | 50.46 | 86.29 | 64.05 | 1264 | 70.81 |
| 2021 | CondLaneNet [53] (ResNet-101) | 58 | 79.48 | 93.47 | 77.44 | 70.93 | 80.91 | 54.13 | 90.16 | 75.21 | 1201 | 74.80 |
| 2022 | UFLDv2 [43] (ResNet-34) | 165 | 75.90 | 92.50 | 74.90 | 65.70 | 75.30 | 49.00 | 88.50 | 70.20 | 1864 | 70.60 |
| 2022 | CLRNet [31] (DLA-34) | 94 | 80.47 | 93.73 | 79.59 | 75.30 | 82.51 | 54.58 | 90.62 | 74.13 | 1155 | 75.37 |
| 2024 | HGLNet [54] (DLA-34) | 104 | 81.83 | 93.96 | 79.78 | 76.20 | 83.27 | 55.89 | 90.83 | 75.77 | 1208 | 76.44 |
| Parameter-based methods | | | | | | | | | | | | |
| 2022 | BézierLaneNet [30] (ResNet-34) | 150 | 75.57 | 91.59 | 73.20 | 69.20 | 76.74 | 48.05 | 87.16 | 62.45 | 888 | 69.90 |
| 2023 | BSNet [27] (DLA-34) | 119 | 80.28 | 93.87 | 78.92 | 75.02 | 82.52 | 54.84 | 90.73 | 74.71 | 1485 | 75.59 |
| 2023 | DecoupleLane [55] (DLA-34) | 110 | 80.82 | 93.85 | 80.32 | 76.31 | 82.34 | 55.40 | 90.48 | 75.12 | 1035 | 75.40 |
| Transformer-based methods | | | | | | | | | | | | |
| 2021 | LSTR [19] (Baseline) | | 68.72 | 86.78 | 67.34 | 56.63 | 59.82 | 40.10 | 78.66 | 56.64 | 1166 | 59.92 |
| 2023 | CondLSTR [37] (ResNet-101) | 45 | 80.77 | 94.17 | 79.90 | 75.43 | 80.99 | 55.00 | 90.97 | 76.87 | 1047 | 75.11 |
| 2024 | LATR [56] (ViT-Base) | 78 | 80.85 | 93.92 | 80.21 | 76.04 | 81.65 | 55.42 | 89.53 | 75.66 | 1043 | 75.81 |

As illustrated in Table 2, Transformer-based models have demonstrated notable advantages in addressing complex road scenarios, especially those involving visual ambiguities or temporal inconsistencies. For instance, recent models such as CondLSTR [37] and LATR [56], both incorporating Transformer architectures, achieve competitive total F1 scores of 80.77 and 80.85, respectively—significantly outperforming earlier Transformer-based attempts like LSTR [19] (68.72). These improvements stem from the enhanced ability of Transformers to model long-range dependencies in both spatial and temporal dimensions. In challenging scenes such as dazzle light, curve, and night, these methods consistently yield higher F1 scores. For example, LATR [56] achieves 76.04 in dazzle and 75.66 in curve scenes, exceeding the performance of many conventional convolutional models. This highlights the importance of temporal-spatial context modeling for robustness in diverse driving conditions.

In addition to accuracy, Transformer-based and attention-augmented models are increasingly achieving real-time inference speeds. LATR [56], for example, runs at 78 FPS while maintaining state-of-the-art accuracy, making it suitable for deployment in time-sensitive applications. Furthermore, attention-based CNN models like HGLNet [54], although not strictly Transformer-based, mirror the benefits of spatial context aggregation through hybrid attention mechanisms. HGLNet [54] attains the highest total F1 score (81.83) while exhibiting strong and balanced performance across scenarios such as night (76.44), curve (75.77), and arrow (90.83). These findings underscore that attention mechanisms and temporal-spatial enhancement modules not only improve robustness under adverse conditions but also allow for efficient inference—addressing two critical demands of modern lane detection systems: accuracy and real-time performance.

# 4. Discussion

## 4.1 Lane representation methods

Table 3 compares the lane representation methods discussed in previous sections, detailing their advantages and disadvantages. It also presents the best-performing model and its score for each lane representation method on the CULane [15] test set. Table 3 shows that, while the parameter-based method achieves the best performance among the three, the score difference is less than 0.2% compared to the lowest-performing method. This indicates that the margin of performance between the methods is tiny. Additionally, it is worth noting that the Lane2Seq [57] model, proposed by Zhou, K. (2024), uses ViT-Base [58] as its backbone, employing a transformer structure to extract

image features and leverage their correlations for lane detection. This model unifies different lane representation methods (such as segmentation, point-based, and parametric methods) into a sequence generation task.

**Table 3:** Comparison of Deep Learning Methods on The CULane [15] Dataset

| Methods & Model (backbone) | $F1_{50}$ score | Advantages | Disadvantages |
|---|---|---|---|
| Segmentation-based, CondLSTR [37] (ResNet-101) | 80.77 | Handling Multi-Lane Scenarios, including lane changes and merges. | Computational Complexity.Sensitivity to Environmental Conditions |
| Point-based, CLRNet [31] (DLA34) | 80.68 | Efficiency in Detection. Robustness to challenging environments. | Fixed-shape anchors cannot flexibly adapt to various lane shapes and curvatures. Require complex post-processing, such as NMS. |
| Parameter-based, DecoupleLane [55] (DLA34) | 80.82 | Offer quicker inference times | The detection results are highly sensitive to the accuracy of the predicted parameters. |

1) Accuracy: Performance differences between the three primary lane representation methods (Segmentation, Point-based, Parameter-based) are remarkably small (< 0.2% $F1_{50}$ difference on CULane [15]). The highest score (80.82) is achieved by the Parameter-based method (DecoupleLane [55]), closely followed by Segmentation (80.77) and Point-based (80.68). Lane2Seq's unified approach yields similar results across methods (79.64, 79.27, 78.39), further confirming that the choice of fundamental representation itself has a limited impact on peak accuracy potential for modern models.
2) Robustness: Each method has distinct robustness strengths and weaknesses.
3) Segmentation: Excels in handling complex multi-lane scenarios (changes, merges) but is sensitive to environmental conditions (e.g., lighting, weather).
4) Point-based: Praised for robustness in challenging environments but struggles with highly variable lane shapes due to fixed anchors and requires complex post-processing (e.g., NMS), potentially impacting robustness to occlusions or extreme curvatures.
5) Parameter-based: Offers efficiency, but its results are highly sensitive to the accuracy of the predicted parameters, potentially making it less robust to noisy input or unusual lane geometries.
6) Runtime: Parameter-based methods are explicitly noted for offering quicker inference times. Segmentation methods often incur higher computational complexity. Point-based methods add computational overhead through complex post-processing steps.

**Table 4:** Overview of Various Models That Integrate Temporal Information for Lane Detection

| Models | Backbone and Fusion Method | Dataset and Findings |
|---|---|---|
| MMA-Net [28] | Backbone: UNet [59]<br>Fusion Method:<br>Concatenation & attention | Dataset: VIL-100 [49]<br>Findings: Improves $F1_{50}$ score (0.839 vs. 0.756, a gain of 8.3%).<br>Number of frames: 7. |
| TGC-Net [45] | Backbone: Encoder - Decoder<br>Fusion Method: temporal recurrent feature-shift aggregation module (T-RESA) | Dataset: VIL-100 [49]<br>Findings: Improves $F1_{50}$ score (0.892 vs. 0.839, a gain of 6.31%).<br>Number of frames: 3. |
| Temporal LaneATT [60] | Backbone: LaneATT [41]<br>Fusion Method: Concatenation | Dataset: VIL-100 [49]<br>Findings: Improves $F1_{50}$ score (0.8466 vs. 0.8232, a gain of 2.34%).<br>using more frames may degrade the results. Additionally, the model does not benefit from incorporating attention features with global information from past frames ($F1_{50}$ score: 0.8466 vs. 0.8419).<br>Number of frames: 3. |
| RVLD [51] | Backbone: Encoder-Decoder<br>Fusion Method:<br>Motion Estimation | Dataset: VIL-100 [49]<br>Findings: Improves $F1_{50}$ score (0.924 vs. 0.839, a gain of 8.5%).<br>Number of frames: 1. |
| LaneTCA [52] | Backbone: Encoder-Decoder<br>Fusion Method:<br>Self–attention [32] | Dataset: VIL-100 [49]<br>Findings: Improves $F1_{50}$ score (0.933 vs. 0.924, a gain of 0.9%).<br>Number of frames: 1. |

## 4.2 Temporal information integration methods

Table 4 provides an overview of various models that integrate temporal information for lane detection. The table compares different methods in terms of their backbone models, fusion techniques, and the datasets used, alongside their performance improvements. From the table, we can observe that the integration of temporal features consistently leads to improvements in detection performance, with models like MMA-Net [49] showing significant gains in $F1_{50}$ scores. The findings suggest that while the number of frames used in temporal fusion varies, models using multiple frames or motion estimation methods tend to perform better. However, some methods, such as Temporal LaneATT [60], indicate that incorporating too many frames or irrelevant temporal features may not always lead to further improvements, highlighting the importance of selecting the right fusion method and the optimal number of frames for a given scenario.

1) Accuracy: Integrating temporal information consistently provides significant accuracy gains across all models. Improvements range from substantial (MMA-Net [49]: +8.3%, RVLD [51]: +8.5%, TGC-Net [45]: +6.31%) to moderate (Temporal LaneATT [60]: +2.34%) and minor (LaneTCA [52]: +0.9%). RVLD [51](92.4%) and LaneTCA [52](93.3%) achieve the highest absolute scores, demonstrating the power of leveraging motion [51] or sophisticated attention [52] over time, even with a few frames.
2) Robustness: Temporal fusion inherently enhances robustness to transient occlusions, motion blur, and rapidly changing lighting/weather by leveraging information from multiple moments. Motion-based methods [51] They are particularly robust to dynamic scenes.
3) Runtime: Utilizing multiple frames inherently increases computational cost and memory requirements compared to single-frame methods. The fusion method complexity also impacts runtime: simple concatenation is faster than motion estimation or recurrent and

attention mechanisms. The optimal number of frames is crucial; using too many (Temporal LaneATT [60]) can degrade performance and unnecessarily increase computation.

### 4.3 Global Contextual Information Integration Methods

Table 5 compares various methods for capturing global contextual information. The results show that methods leveraging FPN and attention mechanisms, such as CondLaneNet [53] and CLRNet [31], perform significantly better in capturing long-range dependencies in the feature map space, leading to improved model performance. These methods effectively integrate multi-scale information and focus on task-relevant features, highlighting the importance of FPN and attention mechanisms in global context capturing for tasks like lane detection. Therefore, the detection performance of the model can be effectively enhanced by constructing long-distance dependencies of features to capture global contextual information.

**Table 5:** Comparison of methods to capture global contextual information

| Model (Backbone) | Methods | $F1_{50}$ Score (CULane [15] dataset) |
|---|---|---|
| SCNN [15] (VGG16) | Slice convolution of different aspects is performed on the feature map (the last layer of the backbone). | 71.6 |
| UFLD [42] (ResNet-34) | Generated by the last layer of the backbone. | 72.3 |
| RESA [18] (ResNet-50) | Based on SCNN, repeatedly shifting slice feature maps in vertical and horizontal directions. | 75.3 |
| LaneATT [41] (ResNet-34) | Anchor-based attention mechanism | 76.68 |
| CondLaneNet [53] (ResNet-34) | Add a Transformer encoder to the last layer of the backbone, establishing FPN from the last four-layer feature maps of the backbone | 78.74 |
| CLRNet [31] (ResNet-34)/ (DLA34) | Establishing FPN from the last three-layer feature maps of the backbone, the ROI attention mechanism | 79.73/ 80.47 |
| DecoupleLane [55] (DLA34) | The self-attention mechanism is used to effectively capture the global features of the image | 80.82 |
| HGLNet [54] (ResNet-34)/ (DLA34) | Combination of global-local decoupled modeling, sparse sampling strategies, and dynamic weight allocation, leveraging self-attention and multi-scale feature aggregation to balance performance and efficiency | 81.23/ 81.83 |

1) Accuracy: Methods effectively capturing global context show a clear positive trend in F1 scores on CULane [15], rising from early methods (SCNN [15]: 71.6, UFLD [42]: 72.3) to state-of-the-art (HGLNet [54]: 81.83). Key advancements driving this are:
2) FPN: Integrating multi-scale features (CondLaneNet, CLRNet [31], HGLNet [54]) provides significant boosts over using only the last layer (SCNN [15], UFLD [42]).
3) Attention Mechanisms: Self-attention [32] (DecoupleLane [55], LaneTCA [52]), anchor-based attention (LaneATT [41]), and ROI attention (CLRNet [31]) are highly effective for modeling long-range dependencies, leading to substantial performance improvements. HGLNet [54] A combination of global-local modeling and multi-scale fusion achieves the highest score.
4) Recursive Feature Refinement: Methods like RESA [18] (improved SCNN [15]) also offers gains.
5) Robustness: Global context is crucial for robustness against occlusions, distant lane detection, complex intersections, and ambiguous lane markings. FPN helps handle scale variations. Attention mechanisms allow the model to focus on relevant lane features across the entire scene, ignoring distractions.
6) Runtime: Simpler context methods (last layer only – UFLD [42]) are computationally lighter but less effective. FPN adds moderate overhead. Attention mechanisms, especially self-attention, can be computationally expensive; however, methods like HGLNet [54] explicitly aim for an efficiency balance. Recursive methods (RESA [18], SCNN [15]) also adds computational cost.

### 4.4 Key drivers in lane detection

A comparative analysis of lane representation methods (segmentation, point-based, parametric), temporal fusion strategies, and global context modeling reveals that the choice of fundamental lane representation has minimal impact (<0.2% difference) on peak accuracy for state-of-the-art models in benchmark datasets like CULane [15]. Significant performance gains are primarily driven by two factors:

1) Effective Temporal Fusion: Integrating temporal information via motion estimation (e.g., RVLD [51]) or attention mechanisms (e.g., LaneTCA [52]) yields substantial F1-score ($F1_{50}$) improvements (up to +8.5%), even with only 1-3 frames, far exceeding gains from simple concatenation or increased frame counts.
2) Powerful Global Context Modeling: Combining FPN with attention mechanisms (e.g., CLRNet [31], DecoupleLane [55], HGLNet [54]) significantly enhances long-range dependency capture, boosting $F1_{50}$ scores by 10% compared to early methods (e.g., SCNN [15] → HGLNet [54]).

These techniques concurrently enhance model robustness: temporal fusion mitigates transient issues (occlusion, motion blur), and global context addresses spatial challenges (occlusion, complex layouts). while complex temporal fusion (especially multi-frame) and advanced global modeling (e.g., self-attention) demand higher resources. Optimal model design (e.g., RVLD [51], LaneTCA [52], HGLNet) requires meticulous balancing of performance, robustness, and efficiency, emphasizing the quality of fusion methods and context modeling over simply adding complexity or frames.

## 5. Future works

Current lane detection systems exhibit three critical limitations. First, temporal fusion methods (e.g., MMA-Net [49], TGC-Net [45]) rely on fixed historical frames (3–7 frames), leading to redundant computations or insufficient context utilization when scene dynamics vary (Table II). Second, global context modeling techniques (e.g., HGLNet [54], CLRNet [31]) incur high computational overhead due to

exhaustive attention mechanisms or FPN architectures while struggling with irrelevant spatial noise (Table III). Third, spatial and temporal features are typically fused late (e.g., concatenation) [39], [52], failing to capture synergistic spatiotemporal relationships essential for co-occurring challenges like fog and motion blur. These limitations restrict the balance between model robustness and efficiency, as seen in either minor performance improvements (e.g., Temporal LaneATT [60] only achieving a +2.34% $F1_{50}$ gain) or excessive resource consumption (as detailed in Tables I–III). To overcome these challenges, three key research directions will be explored.

1) Dynamic Temporal Fusion: An adaptive frame selection mechanism will be developed to dynamically adjust the number of fused frames (1–5) based on real-time scene dynamics (e.g., occlusion ratio, motion blur severity). This leverages differentiable motion distillation to compress multi-frame features into lightweight latent representations, enhancing stability (e.g., transient occlusion handling) while maintaining RVLD-level efficiency.
2) Sparse Global Context Modeling: Topology-aware sparse attention replaces full-image self-attention by constraining computations to lane-connected regions via a highly efficient global context modeling module (e.g., graph neural networks, attention-based aggregators, or other structured models) to prioritize relevant spatial dependencies while reducing redundant computations dynamically. Combined with an implicit parametric point generator (B-spline-based), this eliminates manual anchor tuning and NMS post-processing while improving accuracy in extreme curvature or poor-visibility scenarios.
3) Unified Spatiotemporal Embedding: A cross-scale spatiotemporal transformer will be developed to integrate motion dynamics from historical frames and multi-resolution spatial features (augmented via FPN) into a unified spatiotemporal embedding. This architecture incorporates Challenge-aware gating mechanisms to dynamically modulate the relative importance of spatial and temporal features in response to real-time environmental perturbations. For instance, the gating strategy upweights temporal cues during occlusion events (e.g., transient obstructions caused by dynamic agents such as vehicles or pedestrians) and amplifies spatial discriminability under adverse visibility conditions (e.g., atmospheric turbulence, precipitation, or sensor glare). By contextually adapting feature contributions, the model achieves generalizable robustness in complex, co-occurring environmental challenges (e.g., concurrent fog and motion artifacts), thereby mitigating reliance on dataset-specific tuning.
4) Deployment-Oriented Model Optimization: While Transformer-based models (e.g., CondLSTR [37], LATR) achieve strong performance across challenging conditions, their high computational complexity poses barriers to real-world deployment, especially in resource-constrained environments. Future work will investigate hardware-efficient variants, including lightweight attention modules, dynamic query pruning, and hardware-aware NAS (Neural Architecture Search), to facilitate inference on edge devices such as automotive-grade GPUs. This includes targeting high FPS (>50) with minimal memory footprint while preserving robustness in complex scenes (Table II), making large-scale deployment in ADAS and embedded platforms more feasible.
5) Cross-Domain Implications and Standardization Impact: In addition to technical improvements, future research will explore how robust lane detection contributes to broader industrial and policy contexts. Accurate and consistent detection under extreme scenarios (e.g., dazzle, occlusion, low light) could inform the development of standardized testing protocols for perception modules. These standards can serve as benchmarks for regulatory certification of ADAS features and autonomous driving systems. Furthermore, insights gained from temporal-spatial context modeling may guide interdisciplinary collaborations to shape automotive safety regulations, inform public dataset design, and promote reproducibility and fairness in algorithmic benchmarking.
6) Ethical and Generalization Considerations: Although current lane detection benchmarks, such as CULane [15] and VIL-100 [49] provide diverse traffic scenarios, they are still region-specific and may not fully represent global variations in road conditions and lane markings. This introduces potential biases in model training and limits generalizability across geographic regions. To enhance fairness and robustness, future research should consider constructing training datasets with greater geographical diversity and road style variations. In addition, integrating domain generalization or domain adaptation techniques may help mitigate distributional shifts, enabling models to better handle unseen environments and ensure equitable performance across countries and infrastructure types.

## 6. Conclusion

This comprehensive review has traced the evolution of deep learning-based lane detection methodologies, emphasizing their capacity to address the complexities inherent in dynamic driving environments. Deep learning-based lane detection methods demonstrate marked advantages over traditional image-processing techniques such as Hough transforms and edge detection. However, performance differences among the three primary lane representation paradigms—segmentation-based, point-based, and parameter-based approaches—remain remarkably small, with $F1_{50}$ disparities below 0.2% on the CULane [15] benchmark dataset. This observation highlights a critical limitation: relying solely on lane representation strategies is insufficient to address fundamental challenges such as occlusions, illumination variability, structural ambiguities, and multimodal sensor interference. Two architectural innovations have emerged as pivotal for enhancing robustness. First, temporal feature fusion leverages sequential frame data through motion estimation (e.g., in RVLD [51]) or attention mechanisms (e.g., in LaneTCA [52])—to mitigate transient occlusions and dynamic distortions, achieving improvements of up to 8.5% in $F1_{50}$. Second, global context modeling integrates FPN with attention mechanisms (as seen in CLRNet [31] and HGLNet [54]), enabling the capture of long-range spatial dependencies. This approach reduces errors caused by distant lane ambiguities or fragmented detections by approximately 10% compared to earlier methods.

Despite these advancements, practical deployment constraints persist. Temporal fusion and global context modules frequently incur prohibitive computational costs, while sensitivity to extreme environmental multimodality (e.g., simultaneous fog, motion blur, and adverse lighting) remains unresolved. Furthermore, late-stage fusion of spatial and temporal features fails to exploit synergistic relationships critical for co-occurring challenges. These gaps necessitate three transformative research directions: dynamic adaptive frameworks capable of modulating temporal context (e.g., occlusion-aware frame sampling) and compressing multi-frame features efficiently; sparse topology-aware modeling that replaces exhaustive attention with lane-centric sparse computations (e.g., graph neural networks) to balance accuracy and latency; and unified spatiotemporal embeddings, such as cross-scale transformers with challenge-aware gating, that holistically integrate motion dynamics and multi-resolution spatial features for robustness under compound adversities. Achieving reliable lane detection in unstructured real-world environments demands rigorous co-optimization of robustness, computational efficiency, and cross-scenario generalizability.

## Acknowledgement

This work is funded by the Research Fund of Nanning University.

## References


[1] Y. Zhang, Z. Tu, and F. Lyu, "A Review of Lane Detection Based on Deep Learning Methods," Mech. Eng. Sci., vol. 5, no. 2, May 2024, doi: 10.33142/mes.v5i2.12721.

[2] Q. Zou, H. Jiang, Q. Dai, Y. Yue, L. Chen, and Q. Wang, "Robust Lane Detection from Continuous Driving Scenes Using Deep Neural Networks," IEEE Trans. Veh. Technol., vol. 69, no. 1, pp. 41–54, Jan. 2020, doi: 10.1109/TVT.2019.2949603.

[3] Department of Computer Science and Engineering, Khulna University of Engineering & Technology, Khulna-9203, Bangladesh, Md. Rezwanul Haque, Md. Milon Islam, K. Saeed Alam, and H. Iqbal, "A Computer Vision based Lane Detection Approach," Int. J. Image Graph. Signal Process., vol. 11, no. 3, pp. 27–34, Mar. 2019, doi: 10.5815/ijigsp.2019.03.04.

[4] J. Canny, "A Computational Approach to Edge Detection," IEEE Trans. Pattern Anal. Mach. Intell., vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986, doi: 10.1109/TPAMI.1986.4767851.

[5] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," Commun ACM, vol. 15, no. 1, pp. 11–15, 1972, doi: 10.1145/361237.361242.

[6] X. He et al., "Monocular Lane Detection Based on Deep Learning: A Survey," Dec. 11, 2024, arXiv: arXiv:2411.16316. doi: 10.48550/arXiv.2411.16316.

[7] L. Deng, H. Cao, and Q. Lan, "Dynamically Enhanced lane detection with multi-scale semantic feature fusion," Comput. Electr. Eng., vol. 118, p. 109426, Sep. 2024, doi: 10.1016/j.compeleceng.2024.109426.

[8] V. Devane, G. Sahane, H. Khairmode, and G. Datkhile, "Lane Detection Techniques using Image Processing," ITM Web Conf., vol. 40, p. 03011, 2021, doi: 10.1051/itmconf/20214003011.

[9] S. Sultana, B. Ahmed, M. Paul, M. R. Islam, and S. Ahmad, "Vision-Based Robust Lane Detection and Tracking in Challenging Condi-tions," IEEE Access, vol. 11, pp. 67938–67955, 2023, doi: 10.1109/ACCESS.2023.3292128.

[10] R. K. Megalingam, N. C. Pradeep, A. Reghu, S. A. Sreemangalam, A. Ayaaz, and A. Hegde Kota, "Lane Detection Using Hough Trans-form and Kalman Filter," in 2024 International Conference on E-mobility, Power Control and Smart Systems (ICEMPS), Thiruvanan-thapuram, India: IEEE, Apr. 2024, pp. 01–05. doi: 10.1109/ICEMPS60684.2024.10559324.

[11] Y. Zhang, Z. Lu, X. Zhang, J.-H. Xue, and Q. Liao, "Deep Learning in Lane Marking Detection: A Survey," IEEE Trans. Intell. Transp. Syst., vol. 23, no. 7, pp. 5976–5992, Jul. 2022, doi: 10.1109/TITS.2021.3070111.

[12] N. Sukumar and P. Sumathi, "A Robust Vision-based Lane Detection using RANSAC Algorithm," in 2022 IEEE Global Conference on Computing, Power and Communication Technologies (GlobConPT), New Delhi, India: IEEE, Sep. 2022, pp. 1–5. doi: 10.1109/GlobConPT57482.2022.9938320.

[13] U. Khamdamov, A. Abdullayev, M. Mukhiddinov, and S. Xalilov, "Algorithms of Multidimensional Signals Processing based on Cubic Basis Splines for Information Systems and Processes," J. Appl. Sci. Eng., vol. 24, no. 2, pp. 141–150, 2021, doi: 10.6180/jase.202104_24(2).0003.

[14] D. Neven, B. D. Brabandere, S. Georgoulis, M. Proesmans, and L. V. Gool, "Towards End-to-End Lane Detection: an Instance Seg-mentation Approach," Feb. 15, 2018, arXiv: arXiv:1802.05591. doi: 10.48550/arXiv.1802.05591.

[15] X. Pan, J. Shi, P. Luo, X. Wang, and X. Tang, "Spatial As Deep: Spatial CNN for Traffic Scene Understanding," Dec. 17, 2017, arXiv: arXiv:1712.06080. Accessed: Jul. 29, 2024. [Online]. Available: http://arxiv.org/abs/1712.06080

[16] Y. Li et al., "Nighttime lane markings recognition based on Canny detection and Hough transform," in 2016 IEEE International Confer-ence on Real-time Computing and Robotics (RCAR), Jun. 2016, pp. 411–415. doi: 10.1109/RCAR.2016.7784064.

[17] H. Lyu, Z. Zhu, and S. Fu, "ENet-SAD–A CNN-based lane detection for recognizing various road conditions," in Third International Conference on Algorithms, Network, and Communication Technology (ICANCT 2024), SPIE, Mar. 2025, pp. 268–276. doi: 10.1117/12.3060154.

[18] T. Zheng et al., "RESA: Recurrent Feature-Shift Aggregator for Lane Detection," Mar. 25, 2021, arXiv: arXiv:2008.13719. doi: 10.48550/arXiv.2008.13719.

[19] R. Liu, Z. Yuan, T. Liu, and Z. Xiong, "End-to-end Lane Shape Prediction with Transformers," in 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA: IEEE, Jan. 2021, pp. 3693–3701. doi: 10.1109/WACV48630.2021.00374.

[20] Y. Dong, S. Patil, B. Van Arem, and H. Farah, "A hybrid spatial–temporal deep learning architecture for lane detection," Comput.-Aided Civ. Infrastruct. Eng., vol. 38, no. 1, pp. 67–86, Jan. 2023, doi: 10.1111/mice.12829.

[21] C. Steger, "An unbiased detector of curvilinear structures," IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 2, pp. 113–125, Feb. 1998, doi: 10.1109/34.659930.

[22] J. He, S. Sun, D. Zhang, G. Wang, and C. Zhang, "Lane Detection for Track-following Based on Histogram Statistics," in 2019 IEEE International Conference on Electron Devices and Solid-State Circuits (EDSSC), Xi'an, China: IEEE, Jun. 2019, pp. 1–2. doi: 10.1109/EDSSC.2019.8754094.

[23] S. Annadurai, Fundamentals of Digital Image Processing. Pearson Education India, 2007.

[24] X. Zhang and X. Zhu, "Autonomous path tracking control of intelligent electric vehicles based on lane detection and optimal preview method," Expert Syst. Appl., vol. 121, pp. 38–48, May 2019, doi: 10.1016/j.eswa.2018.12.005.

[25] L. Han, Y. Tian, and Q. Qi, "Research on edge detection algorithm based on improved sobel operator," MATEC Web Conf., vol. 309, p. 03031, 2020, doi: 10.1051/matecconf/202030903031.

[26] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and au-tomated cartography," Commun. ACM, vol. 24, no. 6, pp. 381–395, Jun. 1981, doi: 10.1145/358669.358692.

[27] [H. Chen, M. Wang, and Y. Liu, "BSNet: Lane Detection via Draw B-spline Curves Nearby," Jan. 17, 2023, arXiv: arXiv:2301.06910. Accessed: Nov. 02, 2024. [Online]. Available: http://arxiv.org/abs/2301.06910

[28] L. Tabelini, R. Berriel, T. M. Paixão, C. Badue, A. F. De Souza, and T. Oliveira-Santos, "PolyLaneNet: Lane Estimation via Deep Poly-nomial Regression," Jul. 14, 2020, arXiv: arXiv:2004.10924. Accessed: Jul. 27, 2024. [Online]. Available: http://arxiv.org/abs/2004.10924

[29] R. Raguram, J.-M. Frahm, and M. Pollefeys, "A Comparative Analysis of RANSAC Techniques Leading to Adaptive Real-Time Ran-dom Sample Consensus," in Computer Vision – ECCV 2008, vol. 5303, D. Forsyth, P. Torr, and A. Zisserman, Eds., in Lecture Notes in Computer Science, vol. 5303. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 500–513. doi: 10.1007/978-3-540-88688-4_37.

[30] Z. Feng, S. Guo, X. Tan, K. Xu, M. Wang, and L. Ma, "Rethinking efficient lane detection via curve modeling," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 17062–17070. Accessed: May 19, 2025. [Online]. Available: http://openac-cess.thecvf.com/content/CVPR2022/html/Feng_Rethinking_Efficient_Lane_Detection_via_Curve_Modeling_CVPR_2022_paper.html

[31] T. Zheng et al., "CLRNet: Cross Layer Refinement Network for Lane Detection," in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA: IEEE, Jun. 2022, pp. 888–897. doi: 10.1109/CVPR52688.2022.00097.

[32] A. Vaswani et al., "Attention is All you Need," in Advances in Neural Information Processing Systems, Curran Associates, Inc., 2017. Accessed: Jun. 26, 2025. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[33] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image Segmentation Using Deep Learning: A Sur-vey," IEEE Trans. Pattern Anal. Mach. Intell., vol. 44, no. 7, pp. 3523–3542, Jul. 2022, doi: 10.1109/TPAMI.2021.3059968.

[34] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmenta-tion," Jun. 07, 2016, arXiv: arXiv:1606.02147. doi: 10.48550/arXiv.1606.02147.

[35] H. Xu, S. Wang, X. Cai, W. Zhang, X. Liang, and Z. Li, "CurveLane-NAS: Unifying Lane-Sensitive Architecture Search and Adaptive Point Blend-ing," Jul. 23, 2020, arXiv: arXiv:2007.12147. doi: 10.48550/arXiv.2007.12147.

[36] H. Abualsaud, S. Liu, D. Lu, K. Situ, A. Rangesh, and M. M. Trivedi, "LaneAF: Robust Multi-Lane Detection with Affinity Fields," Aug. 20, 2021, arXiv: arXiv:2103.12040. doi: 10.48550/arXiv.2103.12040.

[37] Z. Chen, Y. Liu, M. Gong, B. Du, G. Qian, and K. Smith-Miles, "Generating Dynamic Kernels via Transformers for Lane Detection," in 2023 IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France: IEEE, Oct. 2023, pp. 6812–6821. doi: 10.1109/ICCV51070.2023.00629.
[38] R. Girshick, "Fast R-CNN," Sep. 27, 2015, arXiv: arXiv:1504.08083. doi: 10.48550/arXiv.1504.08083.
[39] X. Li, J. Li, X. Hu, and J. Yang, "Line-CNN: End-to-End Traffic Line Detection With Line Proposal Unit," IEEE Trans. Intell. Transp. Syst., vol. 21, no. 1, pp. 248–258, Jan. 2020, doi: 10.1109/TITS.2019.2890870.
[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.
[41] L. Tabelini, R. Berriel, T. M. Paixão, C. Badue, A. F. D. Souza, and T. Oliveira-Santos, "Keep your Eyes on the Lane: Real-time Atten-tion-guided Lane Detection," Nov. 18, 2020, arXiv: arXiv:2010.12035. doi: 10.48550/arXiv.2010.12035.
[42] Z. Qin, H. Wang, and X. Li, "Ultra Fast Structure-aware Deep Lane Detection," Aug. 04, 2020, arXiv: arXiv:2004.11757. Accessed: Jul. 30, 2024. [Online]. Available: http://arxiv.org/abs/2004.11757
[43] Z. Qin, P. Zhang, and X. Li, "Ultra Fast Deep Lane Detection with Hybrid Anchor Driven Ordinal Classification," Jun. 15, 2022, arXiv: arXiv:2206.07389. Accessed: Jul. 31, 2024. [Online]. Available: http://arxiv.org/abs/2206.07389
[44] L. Chen et al., "PersFormer: 3D Lane Detection via Perspective Transformer and the OpenLane Benchmark," Jul. 19, 2022, arXiv: arXiv:2203.11089. doi: 10.48550/arXiv.2203.11089.
[45] M. Wang, Y. Zhang, W. Feng, L. Zhu, and S. Wang, "Video Instance Lane Detection via Deep Temporal and Geometry Consistency Constraints," in Proceedings of the 30th ACM International Conference on Multimedia, Lisboa Portugal: ACM, Oct. 2022, pp. 2324–2332. doi: 10.1145/3503161.3547914.
[46] L. Zhuang, T. Jiang, M. Qiu, A. Wang, and Z. Huang, "Transformer Generates Conditional Convolution Kernels for End-to-End Lane Detection," IEEE Sens. J., vol. 24, no. 17, pp. 28383–28396, Sep. 2024, doi: 10.1109/JSEN.2024.3430234.
[47] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI: IEEE, Jul. 2017, pp. 936–944. doi: 10.1109/CVPR.2017.106.
[48] J. Han et al., "Laneformer: Object-Aware Row-Column Transformers for Lane Detection," Proc. AAAI Conf. Artif. Intell., vol. 36, no. 1, pp. 799–807, Jun. 2022, doi: 10.1609/aaai.v36i1.19961.
[49] Y. Zhang et al., "VIL-100: A New Dataset and A Baseline Model for Video Instance Lane Detection," in 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada: IEEE, Oct. 2021, pp. 15661–15670. doi: 10.1109/ICCV48922.2021.01539.
[50] Z. Qiu, J. Zhao, and S. Sun, "MFIALane: Multiscale Feature Information Aggregator Network for Lane Detection," IEEE Trans. Intell. Transp. Syst., vol. 23, no. 12, pp. 24263–24275, Dec. 2022, doi: 10.1109/TITS.2022.3195742.
[51] D. Jin, D. Kim, and C.-S. Kim, "Recursive Video Lane Detection," Aug. 22, 2023, arXiv: arXiv:2308.11106. doi: 10.48550/arXiv.2308.11106.
[52] K. Zhou, L. Li, W. Zhou, Y. Wang, H. Feng, and H. Li, "LaneTCA: Enhancing Video Lane Detection with Temporal Context Aggrega-tion," Aug. 25, 2024, arXiv: arXiv:2408.13852. doi: 10.48550/arXiv.2408.13852.
[53] L. Liu, X. Chen, S. Zhu, and P. Tan, "CondLaneNet: a Top-to-down Lane Detection Framework Based on Conditional Convolution," Feb. 10, 2023, arXiv: arXiv:2105.05003. Accessed: Nov. 02, 2024. [Online]. Available: http://arxiv.org/abs/2105.05003
[54] Q. Chang and Y. Tong, "A Hybrid Global-Local Perception Network for Lane Detection," Proc. AAAI Conf. Artif. Intell., vol. 38, no. 2, pp. 981–989, Mar. 2024, doi: 10.1609/aaai.v38i2.27858.
[55] W. Han and J. Shen, "Decoupling the Curve Modeling and Pavement Regression for Lane Detection," Sep. 19, 2023, arXiv: arXiv:2309.10533. Accessed: Jul. 25, 2024. [Online]. Available: http://arxiv.org/abs/2309.10533
[56] Z. Lv, D. Han, W. Wang, and D. Z. Chen, "A Siamese Transformer with Hierarchical Refinement for Lane Detection".
[57] K. Zhou, "Lane2Seq: Towards Unified Lane Detection via Sequence Generation," Feb. 26, 2024, arXiv: arXiv:2402.17172. Accessed: Jun. 11, 2024. [Online]. Available: http://arxiv.org/abs/2402.17172
[58] A. Dosovitskiy et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," Jun. 03, 2021, arXiv: arXiv:2010.11929. Accessed: Jun. 24, 2024. [Online]. Available: http://arxiv.org/abs/2010.11929
[59] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," May 18, 2015, arXiv: arXiv:1505.04597. Accessed: Aug. 07, 2024. [Online]. Available: http://arxiv.org/abs/1505.04597
[60] L. Tabelini, R. Berriel, A. F. De Souza, C. Badue, and T. Oliveira-Santos, "Lane Marking Detection and Classification using Spatial-Temporal Feature Pooling," in 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy: IEEE, Jul. 2022, pp. 1–7. doi: 10.1109/IJCNN55064.2022.9892478.