Volume: 14 Issue: IV Month of publication: April 2026

DOI: https://doi.org/10.22214/ijraset.2026.80005

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Real-Time Lane Detection for Autonomous Driving

Ritu Kumar Jha, Vishal Kumar
Department of Data Science and Design Greater Noida Institute of Technology Greater Noida, India

Abstract: Lane detection stands as a foundational capability for modern autonomous vehicles and advanced driver assistance systems (ADAS). Despite decades of progress, the problem is far from solved — road environments remain unpredictable, lane markings vary dramatically across geographies, and real-time constraints impose strict computational budgets on embedded hardware. This paper offers a structured review of the state of the art in deep learning-based lane detection, tracing the field from early convolutional segmentation approaches through the latest transformer-based architectures. We examine the progression from pixel-level classification frameworks to anchor-driven regression models, polynomial curve fitting strategies, and end-to-end detection transformers. We further survey the benchmark datasets and evaluation metrics that shape how the community measures progress, and critically assess persistent challenges including adverse weather, occlusion, non-standard lane markings, and the sim-to-real gap. Drawing on these findings, we identify directions that are likely to define the next generation of lane detection systems, including multi-task learning, domain-adaptive training, 3D lane reconstruction, and lightweight deployment on embedded automotive platforms.
Keywords: lane detection, autonomous driving, convolutional neural network, transformer, SCNN, CLRNet, encoder-decoder, real-time inference, ADAS, CULane, TuSimple

I. INTRODUCTION

The prospect of vehicles navigating roads without human intervention has moved from science fiction to engineering reality over the past decade. Central to this transition is the ability of a vehicle to continuously understand where it is within a lane and how the road ahead is structured. Lane detection — identifying the painted or physical boundaries that divide traffic lanes — is one of the oldest problems in computer vision applied to driving, and yet it remains one of the most actively researched. The reasons are straightforward: the task is harder than it looks.

On a well-maintained highway under bright sunlight, almost any edge-based algorithm can find white dashed markings with acceptable reliability. The difficulty compounds quickly when the sun angle creates glare, when rain obscures markings behind water films, when vehicles crowd the camera's field of view, or when urban intersections present a confusing mosaic of arrows, pedestrian crossings, and faded paint. Add the fact that a production system must run at thirty frames per second or faster on hardware that draws only a few watts, and the scope of the problem becomes apparent.

Early approaches to lane detection relied on hand-engineered features. The Hough transform, introduced to detect linear structures in images, was one of the earliest tools adapted for lane finding [1]. RANSAC-based fitting helped handle outliers, and Kalman filtering provided temporal smoothing across video frames [2]. These methods worked acceptably in constrained environments but buckled under the diversity of real-world roads. They required tuning of thresholds that rarely generalized, and they offered no natural path toward handling complex curved lanes, partial occlusions, or non-standard marking colors.

The rise of deep learning transformed the field. Convolutional neural networks brought representation learning — the ability to derive features directly from data rather than from human intuition — and this advantage proved decisive. By around 2016, CNN-based methods were outperforming hand-crafted approaches on every major benchmark, and the gap has only widened. Today, the frontier sits at architectures that borrow ideas from natural language processing (attention mechanisms, transformers), from point cloud processing (set-invariant encoders), and from multi-task learning (sharing a backbone across detection, segmentation, and depth estimation simultaneously).

This review is organized as follows. Section 2 provides background on the lane detection problem and the properties of lane markings that make it challenging. Section 3 reviews the evolution of deep learning approaches in roughly chronological order. Section 4 surveys the standard benchmarks and metrics. Section 5 analyzes the open challenges that current methods still struggle with. Section 6 discusses emerging research directions. Section 7 concludes.

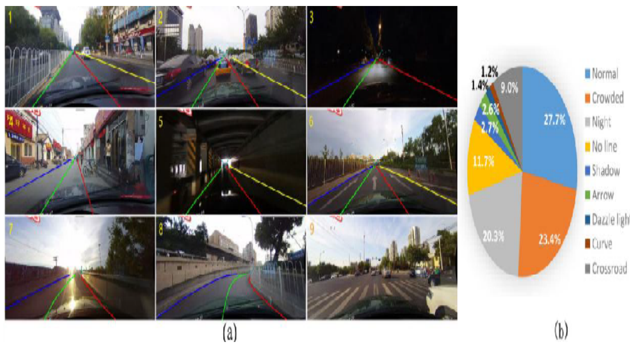
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

II. PROBLEM DEFINITION AND CORE CHALLENGES

A. What Lane Detection Requires

A lane detection system takes one or more camera images as input and produces, as output, the geometric description of every visible lane boundary in the scene. The geometry may be expressed as a set of pixel-level binary masks (one per lane), a collection of key-point coordinates, a set of parametric curves (polynomials, splines, Bezier curves), or — increasingly — as direct 3D world-space representations. Each representation carries its own trade-offs between accuracy, computational cost, and the ease with which downstream modules (path planning, lane-keeping control) can consume the output.

The number of lanes visible in a front-view camera image is not fixed. Highway images may contain two or three lanes in each direction, while complex urban intersections can present far more markings, including turn bays, merge zones, and transverse markings that are irrelevant to lane-following. A robust detector must handle variable output cardinality without being explicitly told how many lanes to look for.



B. Properties That Make Detection Hard

Several properties of lane markings work against detection algorithms. First, markings occupy a small fraction of any image — they are thin, elongated structures without the distinctive texture or color contrast that makes other objects (vehicles, pedestrians) relatively easy to find. Second, their local appearance is often indistinguishable from other road features: road arrows, crosswalk stripes, guardrail reflectors, and even shadows can mimic the edge profile of a lane marking. Third, the structural coherence of a lane — the fact that it is a long, smooth, geometrically predictable entity — is a powerful prior that traditional local-feature detectors cannot exploit efficiently, but which spatial context models can.

Environmental conditions layer additional difficulty onto the structural problem. Rain creates water films that blur and diffuse markings; bright sunlight creates specular reflections that can wash out white paint; nighttime illumination is patchy and unpredictable; fog reduces contrast at distance. These conditions are not rare edge cases — they are conditions that any production system will encounter routinely.

Occlusion by other vehicles is perhaps the most practically significant challenge. In dense urban traffic, a lane can be entirely hidden beneath trucks or buses for several seconds at a time. A good system must either predict where the lane continues based on visible segments, or use temporal memory across frames to bridge the gap [3].

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

III. EVOLUTION OF DEEP LEARNING APPROACHES

A. Segmentation-Based Methods

The earliest and most intuitive application of deep learning to lane detection cast it as a semantic segmentation problem. Each pixel in the image is classified as belonging to a lane or not, and post-processing (clustering, connected-component analysis, or RANSAC curve fitting) extracts individual lane instances from the resulting binary map. This formulation is natural because segmentation networks were already well-developed — FCN [4], SegNet, and U-Net provided strong baselines — and because pixel-level labels are relatively straightforward to create from road footage.

The key limitation of naive segmentation for lanes is that convolutional filters are local. Standard convolution aggregates information within small spatial neighborhoods, which means that a filter looking at one segment of a lane has no direct access to the structural context of that same lane elsewhere in the image. This local blind spot is exactly the weakness that lane markings exploit: they derive their identity from their long-range spatial continuity.

SCNN (Spatial CNN), introduced by Pan et al. in 2018, addressed this by adding a layer that propagates information sequentially across the image — left to right, right to left, top to bottom, and bottom to top [5]. Within each direction, slice-by-slice message passing effectively allowed the network to reason about the full length of a lane within a single forward pass. The result was a substantial improvement on the CULane benchmark, and SCNN became a widely used baseline for years afterward. Its weakness is speed: sequential message passing is difficult to parallelize on GPU hardware, making SCNN relatively slow for its accuracy level.

Subsequent segmentation work explored self-attention as a more parallelizable alternative for global context. ENet-SAD used self-attention distillation to train a lightweight model that mimicked the global sensitivity of a heavier teacher network. RESA generalized SCNN's directional aggregation into a more flexible recurrent spatial aggregation module. These variants improved speed without completely closing the gap that attention-free models faced when dealing with partially occluded lanes.

B. Anchor-Based and Regression Methods

Rather than classifying pixels, anchor-based methods frame lane detection as a regression problem: predict the geometric parameters of each lane directly. This is analogous to the anchor-box approach in object detection, adapted for the elongated, nearly-vertical geometry of lane markings.

Line-CNN (2019) was among the first to propose line anchors — rays emanating from the image borders — and to train a network to predict offsets and confidence scores for each anchor [6]. LaneATT (2021) extended this by incorporating an attention mechanism that aggregated features along anchor directions, enabling each candidate lane to gather global evidence before making a prediction [7]. The result was a model that was simultaneously faster and more accurate than SCNN on standard benchmarks, achieving good performance even on partially occluded lanes.

CLRNet (Cross Layer Refinement Network, 2022) pushed this paradigm further by introducing cross-layer refinement: high-level semantic features detect approximate lane positions while low-level detail features refine the localization [8]. A specialized ROIGather module aggregates global contextual information around each detected lane candidate. The Line IoU loss, also introduced in CLRNet, treats a lane as a whole unit during regression rather than optimizing point-by-point distances, which substantially improves localization accuracy. CLRNet with a DLA-34 backbone achieved 80.47 F1 on CULane — a significant advance over prior state-of-the-art — while maintaining competitive inference speed.

C. Curve and Polynomial Representation

A parallel strand of research represents each lane as a parametric curve rather than a set of discrete points or segmentation masks. The rationale is that curves are compact, differentiable, and naturally encode the smooth geometry of road lanes.

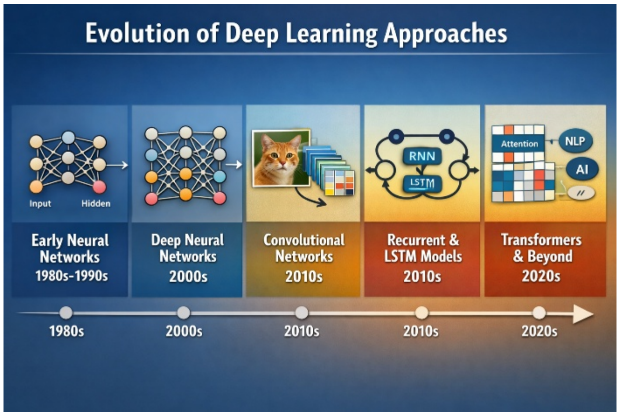
PolyLaneNet (2020) trained a network to predict polynomial coefficients directly from image features, representing each lane as a second- or third-degree polynomial function of image height [9]. This gave an extremely compact representation but struggled with complex lane geometries like S-curves or forking lanes that a low-degree polynomial cannot represent faithfully.

BezierLaneNet (2022) replaced polynomials with Bezier curves, which offer better flexibility and are easier to control numerically [10]. A feature fusion module based on deformable convolution exploited the near-symmetrical structure of lane pairs to improve fitting. BSNet (2023) went further by adopting B-spline curves, which overcome the oscillation problems that arise with high-degree Bezier curves and allow local modifications without affecting the entire curve shape.

D. *Keypoint-Based Methods*

Keypoint detection approaches treat lane localization as the problem of finding a set of characteristic points along each lane and grouping them into lane instances. This is conceptually similar to human pose estimation and benefits from the rich literature of bottom-up and top-down keypoint grouping methods.

CondLaneNet (2021) combined instance conditioning with keypoint prediction, using a conditional convolutional approach to generate instance-specific detection heads [11]. GANet (Graph Aggregation Network) used graph neural networks to model the structural relationships between keypoints, improving performance in scenarios where lane continuity must be inferred across occluded regions.



E. *Transformer-Based End-to-End Methods*

The introduction of DETR (Detection Transformer) to object detection opened the door to end-to-end methods that eliminate post-processing entirely. Applied to lanes, transformer-based detectors learn to directly output lane representations in a set-prediction framework, supervised by bipartite matching losses.

LSTR (Lane Shape Transformer, 2021) was an early adaptation of transformer architectures to lane detection, representing lanes as polynomial parameters decoded by a transformer from image features. The architecture processed the entire image with global self-attention, giving it a natural advantage in reasoning about long-range lane structure.

LDTR (2024) extended this with an anchor-chain lane representation, treating each lane as a sequence of chained points rather than a single polynomial [12]. This representation can handle complex special cases — T-junctions, roundabouts, sharp curves — that polynomial models cannot describe. LDTR incorporated multi-referenced deformable attention and a Gaussian heatmap auxiliary branch to improve convergence and accuracy, demonstrating that transformer architectures could match or exceed anchor-based methods in challenging scenarios.

O2SFormer (2023) addressed a training efficiency problem inherent to DETR-style architectures: one-to-one label assignment creates conflicts that slow convergence [13]. By combining one-to-one and one-to-many assignment strategies during training while maintaining end-to-end inference, O2SFormer achieved faster training convergence without sacrificing the clean, post-processing-free inference pipeline of transformer detectors.

IV. BENCHMARKS AND EVALUATION METRICS

A. *Key Datasets*

The community has produced a range of datasets that differ in geographic coverage, annotation density, and the difficulty of scenes included. The three most widely cited benchmarks in 2D lane detection are TuSimple, CULane, and LLAMAS.

TuSimple was released by TuSimple Inc. and contains roughly 6,000 annotated clips from American highways. Lanes are represented as sequences of y-coordinate-indexed x-positions. The dataset is considered relatively straightforward — highways in good weather with well-maintained markings — and most modern methods achieve accuracy above 96% on it, meaning it functions more as a sanity check than a differentiator.

CULane, collected by cameras mounted on vehicles driving in Beijing, is far more challenging. Its 133,000+ frames cover nine scenario categories including normal, crowded, dazzle light, shadow, no-line, arrow, curve, cross, and night.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Performance on CULane is measured using an intersection-over-union threshold of 0.5 between predicted and annotated lane segments, yielding an F1 score. The 'no-line' category — where the physical lane exists but no marking is visible — is particularly informative as it tests whether a model is doing structural inference or simply finding paint.

LLAMAS provides highly accurate lane annotations derived from HD map ground truth rather than manual labeling, giving it exceptional annotation quality. The dataset covers diverse North American highway driving scenarios. It was introduced partly to address concerns about inter-annotator inconsistency in manually labeled datasets like CULane.

More recently, LanEvil (2024) introduced a robustness benchmark built on the CARLA simulator, with 94 corruption scenarios covering road damage, reflections, shadows, and obstacles [14]. This kind of adversarial benchmark is increasingly important for assessing whether laboratory accuracy translates to real-world reliability.

B. *Evaluation Metrics*

True Positive Rate (TPR, also called recall) and False Positive Rate (FPR) are the standard metrics for evaluating lane detection systems. TPR measures the proportion of annotated lane segments that the model correctly identifies; FPR measures the proportion of predictions that are spurious. These are computed at a fixed IoU threshold — typically 0.5 — between the predicted lane segment and the ground truth.

F1 score (the harmonic mean of precision and recall) is now the dominant summary metric for CULane, because it balances the two types of error. The mean F1 across IoU thresholds from 0.5 to 0.75 (mF1) is used when a more fine-grained view of localization accuracy is needed, penalizing models that find lanes in roughly the right place but with imprecise boundaries.

Frames per second (FPS) is routinely reported alongside accuracy metrics, reflecting the real-time constraint that deployment imposes. A model that achieves 80% F1 but runs at 5 FPS is not deployable, while a model with 78% F1 at 250 FPS may be preferable depending on the hardware platform.

Table 1. Performance comparison of representative lane detection methods on CULane (F1@50) and TuSimple (Accuracy)

Method	Type	CULane F1@50	TuSimple Acc.	Speed (FPS)	Year
SCNN	Segmentation	71.60	96.53%	~7	2018
ENet-SAD	Segmentation	70.80	96.64%	75	2020
LaneATT (R34)	Anchor	77.02	96.86%	77	2021
CondLaneNet (R18)	Keypoint	78.14	97.01%	220	2021
CLRNet (R18)	Anchor+Refine	79.58	97.89%	206	2022
CLRNet (DLA34)	Anchor+Refine	80.47	—	167	2022
BezierLaneNet	Curve	75.57	96.36%	143	2022
LDTR	Transformer	78.16	—	~38	2024
O2SFormer	Transformer	80.25	97.76%	~40	2023

Simultaneously, a Telegram bot API call is issued, sending the crop image and a brief alert message to a designated safety officer channel. A Streamlit dashboard displays a live camera feed with bounding boxes coloured by compliance status (green for Class 0, yellow for Class 1, red for Class 2), a rolling compliance rate chart for the past 60 seconds, and a table of recent violation events. The dashboard runs on the same Raspberry Pi and is accessible on the local network via browser.

V. PERSISTENT CHALLENGES

A. *Adverse Weather and Lighting*

The performance gap between favorable and adverse conditions remains one of the most commercially significant unsolved problems in lane detection. Rain deposits a water film on road surfaces that diffuses the retroreflective paint in lane markings, dramatically reducing contrast. The same rain produces droplets on the camera lens and windshield that blur the entire image in spatially non-uniform ways. Fog attenuates contrast at distance, making far-field lanes invisible. Night removes the ambient illumination that conventional cameras depend on, leaving only the narrow cone of headlights and the occasional streetlamp.

Most standard benchmarks include some representation of these conditions, but the distribution is not representative of what a production vehicle encounters. CULane includes a 'night' split and a 'dazzle' split, but these together account for a relatively small fraction of the test data. A model that performs well on average may still fail unacceptably in the tail of the weather distribution.

The 2D lane detection literature still faces challenges in effectively detecting lane lines under extreme weather conditions such as rain, snow, and fog, as acknowledged by multiple recent surveys [15].

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Addressing this requires both more adversarial training data and architectural choices that are inherently more robust to image degradation — for instance, attention mechanisms that can selectively focus on clearer image regions, or multi-modal approaches that fuse camera data with radar or LiDAR returns that are unaffected by precipitation.

B. Occlusion and Structural Inference

When a large vehicle occupies the lane ahead, the lane markings beneath it are simply not visible in the camera image. A lane detector must either accept that it cannot predict the hidden portion, or infer from context — the visible segments on either side, the predicted motion of the occluding vehicle, temporal memory from previous frames — where the lane most likely continues.

Temporal approaches using recurrent networks or video transformers can bridge short gaps, but they introduce computational overhead and complicate training. Multi-frame approaches also require careful handling of camera motion: the lane in frame t is not at the same pixel coordinates as the same lane in frame t-1 unless ego-motion is properly compensated.

C. Complex Intersection Geometry

Standard lane detection formulations assume that lanes are roughly vertical and continuous across most of the image. Intersections violate both assumptions. Lane markings merge, split, and cross at arbitrary angles; transverse markings (stop lines, crosswalks, turn arrows) add visual clutter that can trigger false positives; and the number and arrangement of relevant markings changes rapidly as the vehicle approaches and enters the intersection.

Transformer-based detectors with flexible lane representations (such as anchor-chains in LDTR) handle some of these cases more naturally than polynomial or anchor-based models, but even they struggle with the most geometrically complex scenarios.

D. Data Annotation Cost and Sim-to-Real Transfer

Training high-quality lane detectors requires large amounts of accurately labeled data. Manual annotation of lane positions, especially for datasets that include quadratic-function fits to each lane, is expensive and time-consuming. Inaccurate annotations propagate directly into model performance.

Simulation offers a path to cheap labeled data — lane positions are known exactly in a simulator — but synthetic imagery differs from real-world cameras in ways that hurt transfer. The domain gap in texture, lighting, and scene composition means that models trained purely on synthetic data often perform significantly worse in the real world. Weakly supervised methods, like the min-distance loss introduced in LaneNet [16], offer a partial solution by allowing networks to be refined using annotations that only specify the number of lanes present rather than their exact positions. Domain adaptation techniques that explicitly reduce the distributional gap between simulation and real-world imagery remain an active research area.

E. Embedded Hardware Constraints

Production automotive hardware is severely resource-constrained relative to the GPU clusters used in research. An NVIDIA Titan Xp delivers around 12 TFLOPS; an embedded platform such as the NVIDIA Jetson Orin delivers roughly 275 TOPS in its highest configuration, but at a much lower power envelope and with different compute trade-offs. Methods that achieve excellent benchmark accuracy on powerful GPU hardware must be significantly compressed, pruned, and quantized before they run at acceptable speed on vehicle-mounted compute.

This hardware reality motivates a recurring tension in the literature: the architectures that currently achieve the highest accuracy (large transformer models, multi-scale feature pyramids) are also the most expensive to run. The fastest methods (row-anchor classifiers, lightweight encoders with depthwise separable convolutions) sacrifice some accuracy for the speed necessary for practical deployment.

VI. EMERGING RESEARCH DIRECTIONS

A. Multi-Task Learning

Lane detection does not need to happen in isolation. A vehicle also needs to detect other vehicles, segment drivable areas, estimate depth, and recognize traffic signs. Running separate networks for each task is expensive; sharing a backbone across tasks is efficient and can improve each task through cross-task regularization.

Multi-task frameworks that simultaneously detect lanes, drivable areas, and objects on the BDD100K dataset have demonstrated that a single efficient encoder can support all three tasks with competitive per-task performance [17]. The key challenge is managing the conflicting gradient signals from tasks with different loss scales and label distributions — lane pixels are rare, making the loss from lane detection inherently smaller than losses from denser tasks.

B. 3D Lane Detection

Two-dimensional lane detection tells a vehicle where lane markings appear in the camera image, but not their true spatial position relative to the vehicle. On flat roads, 2D image coordinates can be converted to approximate 3D ground-plane coordinates through inverse perspective mapping (IPM). But roads are not flat: hills, banked curves, and bridge transitions all violate the flat-ground assumption, and errors in 3D position estimation can accumulate into dangerous trajectory errors.

3D lane detection — directly estimating lane geometry in world-space 3D coordinates — is emerging as a necessary step toward truly reliable autonomous driving [18]. The challenge is that recovering 3D structure from a single camera requires depth estimation, which is ill-posed without additional constraints. Approaches include: learning the ground plane homography implicitly from data, fusing camera data with depth sensors, using multi-camera rigs with stereo geometry, or operating in bird's-eye-view (BEV) representations that simplify 3D reasoning.

C. Foundation Models and Transfer Learning

Large vision-language models pre-trained on internet-scale data have demonstrated remarkable transfer to downstream vision tasks with minimal task-specific supervision. The application of such models to lane detection is still nascent but promising. Features learned from billions of natural images may provide a more robust initialization than ImageNet pre-training, potentially reducing the sensitivity to annotation quality and the severity of the sim-to-real gap.

Lane2Seq (2024) explored representing lane detection as a sequence generation problem, allowing a language-model-style architecture to directly output lane coordinates as token sequences [19]. This framing opens the door to prompting-based adaptation and to leveraging the in-context learning capabilities of large sequence models.

D. Robustness via Adversarial and Synthetic Augmentation

Generative adversarial networks (GANs) and diffusion models can synthesize realistic degraded-weather imagery that would be expensive or dangerous to collect in the real world. Augmenting training data with synthetic rain, fog, night, and glare has been shown to consistently improve performance under those conditions, even when the synthetic images are not photorealistic. Enhanced CycleGAN variants that translate daytime imagery to realistic nighttime scenes have been applied to reduce the domain gap for nighttime detection.

Benchmark datasets like LanEvil formalize this direction by providing standardized corruption scenarios for evaluating robustness, making it possible to track community progress on this front in a rigorous way.

E. Temporal Modeling and Lane Tracking

A video stream provides temporal context that a single-frame detector cannot use. Lanes move smoothly in the image as the vehicle drives; a lane that was visible in the previous frame almost certainly still exists in the current frame, even if momentarily occluded. Recurrent architectures (LSTM, GRU) and video transformers can model this temporal continuity explicitly, making detections more stable and allowing gaps caused by short-term occlusion to be bridged.

Combining detection with explicit tracking — maintaining lane identities across frames and predicting their locations when they are temporarily invisible — is a direction that several recent works have begun to explore. The integration of lane tracking with downstream control systems is a particularly important area, since instabilities in detected lane position translate directly into oscillations in steering output.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

VII. DISCUSSION

Looking across the evolution documented in this review, a consistent pattern emerges: gains in accuracy have been achieved primarily by increasing the spatial scope of reasoning. Hand-crafted methods operated locally. Early CNNs operated within their receptive fields. SCNN extended that scope directionally. Attention mechanisms and transformers extended it globally. The pattern suggests that the next gains will come not from further expanding the spatial scope within a single frame — most modern architectures already attend globally — but from expanding across time (temporal modeling), across modalities (sensor fusion), and across tasks (multi-task learning).

The infrastructure of the field — benchmarks, metrics, open-source implementations — has matured considerably. CULane, LLAMAS, and TuSimple are now standard reference points that allow meaningful comparisons across papers. However, these datasets were collected primarily in China and the United States, under conditions that reflect those geographies. The community needs benchmarks that better represent the diversity of road conditions worldwide: narrow mountain roads, dirt tracks, faded tropical markings, monsoon conditions.

The computational cost of the leading architectures remains a concern for deployment. CLRNet with DLA-34 achieves state-of-the-art accuracy but requires a relatively powerful GPU. Transformer-based methods like LDTR are accurate but run at tens of frames per second on powerful hardware, not hundreds. Closing the gap between what is achievable at research scale and what is deployable on production automotive compute is a critical engineering challenge that the community has not yet fully resolved.

VIII. CONCLUSION

This review has traced the development of deep learning-based lane detection from its origins in semantic segmentation through four broad paradigm shifts: from pixel classification to anchor-driven regression, from single-point representations to curve fitting and keypoint grouping, and most recently to end-to-end transformer architectures that eliminate post-processing entirely. Along the way, the community has produced a rich collection of benchmark datasets and standardized evaluation metrics that allow meaningful comparison across approaches.

The best current methods — CLRNet, LDTR, O2SFormer, and their contemporaries — achieve impressive results on standard benchmarks, but important gaps remain. Adverse weather, complex intersection geometry, the sim-to-real transfer problem, and the tension between accuracy and deployable inference speed all represent open challenges with clear practical significance.

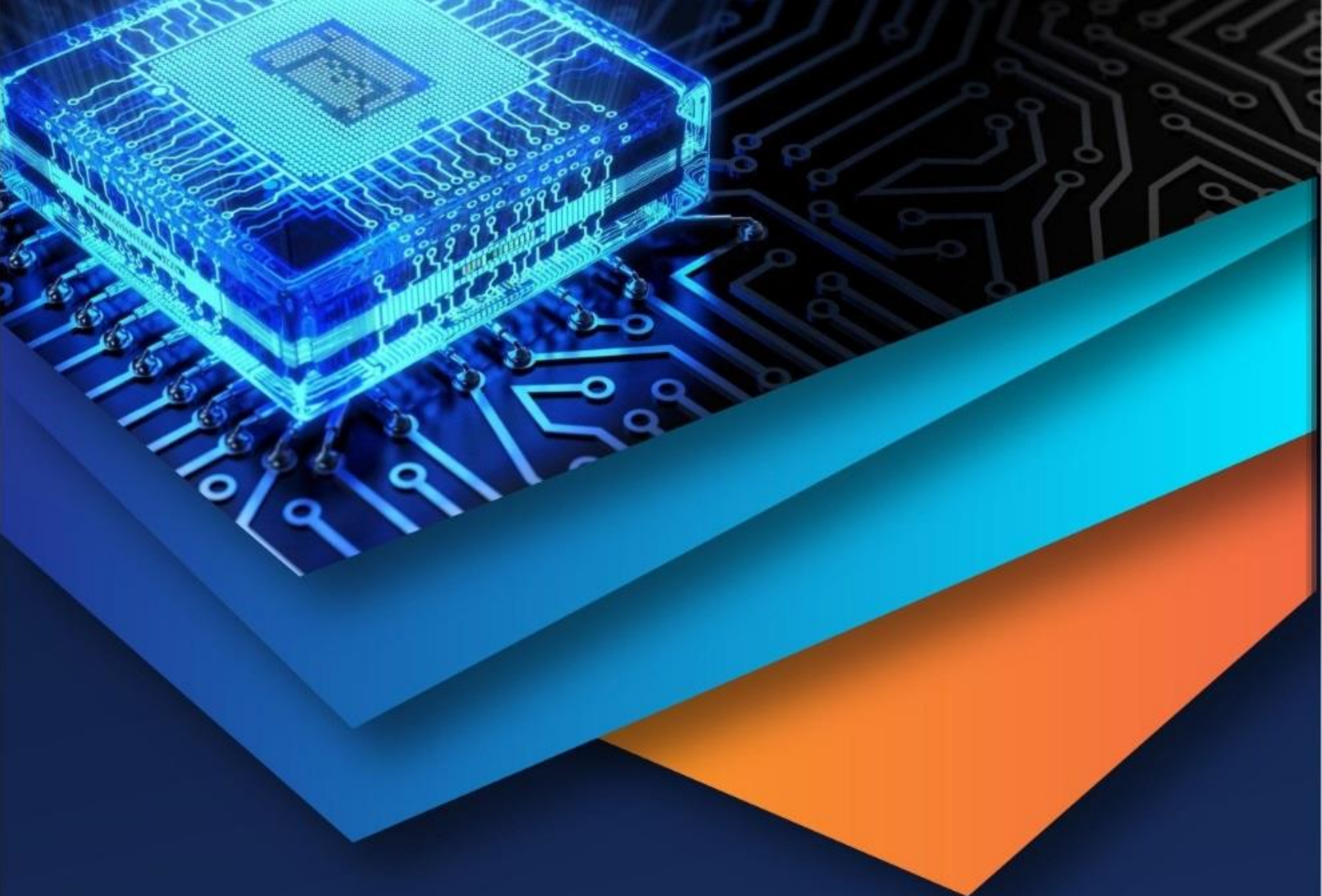
The near-term research agenda is likely to focus on several areas simultaneously: 3D lane detection to handle non-flat roads, multi-task frameworks to improve efficiency, temporal modeling to improve stability, and domain adaptation to close the performance gap between the lab and the real world. Foundation models trained at internet scale may play a growing role as transfer learning substrates, reducing dependence on large domain-specific labeled datasets.

The ultimate measure of progress in lane detection is not a benchmark score but a vehicle that drives safely, consistently, and in conditions that no one thought to include in the training data. That goal remains ahead of us, but the trajectory of progress gives good reason for optimism.

REFERENCES

[1] Duda, R. O., & Hart, P. E. (1972). Use of the Hough transformation to detect lines and curves in pictures. Communications of the ACM, 15(1), 11–15.

[2] Borkar, A., Hayes, M., & Smith, M. T. (2009). Robust lane detection and tracking with RANSAC and Kalman filter. Proceedings of the IEEE International Conference on Image Processing, 3261–3264.
[3] Zou, Q., Jiang, H., Dai, Q., Yue, Y., Chen, L., & Wang, Q. (2019). Robust lane detection from continuous driving scenes using deep neural networks. IEEE Transactions on Vehicular Technology, 69(1), 41–54.
[4] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. Proceedings of the IEEE/CVF CVPR, 3431–3440.
[5] Pan, X., Shi, J., Luo, P., Wang, X., & Tang, X. (2018). Spatial as deep: Spatial CNN for traffic scene understanding. Proceedings of the AAAI Conference on Artificial Intelligence, 32(1).
[6] Li, X., Li, J., Hu, X., & Yang, J. (2019). Line-CNN: End-to-end traffic line detection with line proposal unit. IEEE Transactions on Intelligent Transportation Systems, 21(1), 248–258.
[7] Tabelini, L., Berriel, R., Paixão, T. M., Badue, C., De Souza, A. F., & Oliveira-Santos, T. (2021). Keep your eyes on the lane: Real-time attention-guided lane detection. Proceedings of the IEEE/CVF CVPR, 294–302.
[8] Zheng, T., Huang, Y., Liu, Y., Tang, W., Yang, Z., Cai, D., & He, X. (2022). CLRNet: Cross layer refinement network for lane detection. Proceedings of the IEEE/CVF CVPR, 898–907.
[9] Tabelini, L., Berriel, R., Paixão, T. M., Badue, C., De Souza, A. F., & Oliveira-Santos, T. (2020). PolyLaneNet: Lane estimation via deep polynomial regression. Proceedings of the International Conference on Pattern Recognition (ICPR).
[10] Feng, Z., Guo, S., Tan, X., Xu, K., Wang, M., & Ma, L. (2022). Rethinking efficient lane detection via curve modeling. Proceedings of the IEEE/CVF CVPR, Vancouver.
[11] Liu, L., Chen, X., Zhu, S., & Tan, P. (2021). CondLaneNet: A top-to-down lane detection framework based on conditional convolution. Proceedings of the IEEE/CVF ICCV, 3773–3782.
[12] Yang, Z., Shen, C., Shao, W., et al. (2024). LDTR: Transformer-based lane detection with anchor-chain representation. Computational Visual Media, 10, 753–769.
[13] Zhou, K., & Zhou, R. (2023). End-to-end lane detection with one-to-several transformer (O2SFormer). arXiv preprint arXiv:2305.00675.
[14] Zhang, T., Wang, L., Li, H., et al. (2024). LanEvil: Benchmarking the robustness of lane detection to environmental illusions. Proceedings of the 32nd ACM International Conference on Multimedia, 5403–5412.
[15] Luo, X., Huang, Y., Cui, J., et al. (2025). Deep learning-based lane detection for intelligent driving: A comprehensive survey of methods, datasets, challenges and outlooks. Neurocomputing, ScienceDirect.
[16] Wang, Z., Ren, W., & Qiu, Q. (2018). LaneNet: Real-time lane detection networks for autonomous driving. arXiv preprint arXiv:1807.01726.
[17] Wang, D., Wen, C., Pan, Y., & Shen, C. (2022). End-to-end monocular 3D lane detection via differentiable geometry distillation. Proceedings of the IEEE/CVF ECCV.
[18] He, X., Guo, H., Zhu, K., et al. (2024). Monocular lane detection based on deep learning: A survey. arXiv preprint arXiv:2411.16316.
[19] Cai, Z., et al. (2024). Lane2Seq: Towards unified lane detection via sequence generation. Proceedings of the IEEE/CVF CVPR, 2024.
[20] Maddiralla, V., & Subramanian, S. (2024). Effective lane detection on complex roads with convolutional attention mechanism in autonomous vehicles. Scientific Reports, 14, 19193.



10.22214/IJRASET

INDEX COPERNICUS 45.98

IMPACT FACTOR: 7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY