When Anchors Meet Cold Diffusion: A Multi-Stage Approach to Lane Detection

Bo-Lun Huang[1] Zi-Xiang Ni[1] Feng-Kai Huang[2] Hong-Han Shuai[1*]
Wen-Huang Cheng[2]
[1]National Yang Ming Chiao Tung University
[2]National Taiwan University

Abstract

Accurate and stable lane detection is crucial for the reliability of autonomous driving systems. A core challenge lies in predicting lane positions in complex scenarios, such as curved roads or when markings are ambiguous or absent. Conventional approaches leverage deep learning techniques to extract both high-level and low-level visual features, aiming to achieve a comprehensive understanding of the driving environment. However, these methods often rely on predefined anchors within a single-pass model, limiting their adaptability. The one-shot prediction paradigm struggles with precise lane estimation in challenging scenarios, such as curved roads or adverse conditions like low visibility at night. To address these limitations, we propose a novel cold diffusion-based framework that initializes lane predictions with predefined anchors and iteratively refines them. This approach retains the flexibility and progressive refinement capabilities of diffusion models while overcoming the constraints of traditional hot diffusion techniques. To further enhance the model's coarse-to-fine refinement capabilities, we introduce a multi-resolution image processing strategy, where images are analyzed at different timesteps to capture both global and local lane structure details. Besides, we incorporate a learnable noise variance schedule, enabling the model to dynamically adjust its learning process based on multi-resolution inputs. Experimental results demonstrate that our method significantly improves detection accuracy across a variety of challenging scenarios, outperforming state-of-the-art lane detection methods. Codes and trained weights are available at https://github.com/ntudr/CDiffLane

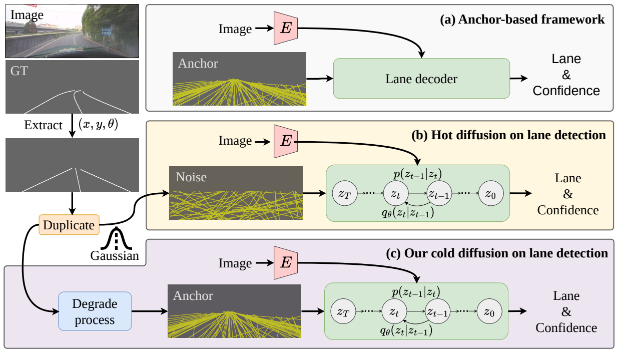

Figure 1. Comparison of various lane detection approaches. (a) Examples of methods such as LaneATT [40], CLRNet [53], and GSENet [39]. (b) Standard diffusion-based approach, similar to DiffusionDet [6], where lane predictions are generated from noisy initializations. (c) Our approach adopts the concept of cold diffusion, beginning the diffusion process from predefined anchors, providing a more reliable starting point for detection.

1. Introduction

Lane detection [16, 32, 33, 43] is a pivotal component of autonomous driving pipelines, as accurately estimating lane boundaries is essential for safe and robust vehicle navigation. Traditional methods [12, 20, 24, 27], often reliant on hand-crafted features and post-processing techniques, require extensive parameter tuning and are sensitive to complex road conditions. Recent advances in deep learning have greatly improved lane detection performance by leveraging convolutional neural networks (CNNs) to extract rich feature representations [4, 26, 28, 53]. Approaches have evolved from segmentation-based methods [26, 28] to anchor-based [4, 53], row-wise detection [31, 32], and parametric regressors [23, 41]. These deep-learning pipelines typically operate in a single forward pass, accelerating inference while improving accuracy.

Despite these strengths, single-pass lane detection methods can struggle under challenging scenarios such as low visibility at night or heavy occlusion, where the one-shot processing often fails to capture the detailed structure of lanes [19, 32]. For instance, lane markings may be partially obscured by other vehicles or faded paint, or harsh lighting conditions, making it difficult for a single forward pass to disambiguate foreground lane pixels from background

[*]Corresponding author.

clutter. Although existing designs incorporate multi-level features, they frequently fuse these features all at once—a practice that overlooks the natural progression from high-level contextual cues (e.g., overall road geometry) down to fine-grained details (e.g., lane boundary edges). As a result, the model lacks a mechanism to iteratively refine uncertain or erroneous predictions. This absence of a hierarchical, coarse-to-fine strategy not only limits the ability to resolve ambiguities in the lane positions but also exacerbates the impact of noise or occlusion in early feature extraction.

A straightforward idea to mitigate the shortcomings is to exploit diffusion models [11, 15, 17, 18, 46, 47], which iteratively refine an initial, coarse prediction by gradually "denoising" it over multiple steps (Figure 1). This iterative mechanism naturally lends itself to self-correction and improved lane delineation. However, directly applying a classic "hot" diffusion process to anchors in lane detection [6] introduces two significant challenges. First, random noise in anchor positions can spread them into unrealistic lane shapes—unlike objects in detection tasks, lanes must follow a predictable structure on the ground plane [19, 22, 32, 40, 53]. Second, simply conditioning on the same image at every timestep fails to exploit a genuine coarse-to-fine mechanism in the image space, causing each iterative step to generate overly similar estimates.

To overcome these issues, we propose *CDiffLane*, a novel framework built on a cold diffusion paradigm that replaces random corruption with a deterministic degradation process tailored for lane anchors. Rather than injecting noise that can drive anchor positions astray, our method progressively transitions from predefined lane anchors toward more refined targets, preserving strong lane priors throughout the diffusion (Figure 1(c)). Besides, we address the problem of repeated conditioning by introducing multi-resolution inputs, enabling our model to progress from lower-resolution, shape-focused representations to higher-resolution details. Finally, we adopt a learnable α schedule in place of standard variance schedules, allowing our system to dynamically control how much each refinement step adjusts the lane anchor position. These enhancements collectively form an efficient, iterative coarse-to-fine pipeline for both training and inference, yielding lane predictions with higher accuracy (particularly under tighter IoU thresholds) and fewer false positives in cross-scenario tests.

We demonstrate the effectiveness of CDiffLane on multiple benchmarks. On the challenging CULane dataset [28], our method achieves substantial gains over state-of-the-art baselines in F_1@75 and shows notable stability improvements across diverse road conditions. These results underscore the robustness and adaptability of CDiffLane, confirming that a careful blend of cold diffusion, anchored priors, and a progressive conditioning strategy can push the boundaries of lane detection performance.

Our contributions are summarized as follows:

- We are the first, to the best of our knowledge, to leverage a generative diffusion model for lane detection, using deterministic degradations guided by empirical lane priors.
- We introduce resolution scheduling and a learnable α that exploit the diffusion process, enabling the model to emphasize different feature levels at each iterative step.
- Our method demonstrates remarkable stability and outperforms state-of-the-art baselines on both CULane and TuSimple [1] datasets, particularly under higher F_1 thresholds and in challenging cross-scenario evaluations.

2. Related Work

2.1. Lane Detection

The methods for lane detection can be categorized into segmentation-based, parameter-based and anchor-based approaches. In segmentation-based methods [29, 51, 52], lane lines are first predicted as dense pixel masks and then fitted into continuous curves, often requiring post-processing to enforce geometric consistency. Parameter-based methods [9, 23, 41] represent lanes using a set of curve parameters, enabling fast inference but exhibiting low tolerance to parameter errors. Meanwhile, anchor-based methods [19, 22, 32, 40, 53] rely on predefined line or row anchors and regress offsets to match lane shapes. By leveraging priors on lane geometry, they can isolate relevant features more efficiently—typically through an IoU-based loss that aligns predicted anchors with ground truth. For instance, Line-CNN [19] and LaneATT [40] incorporate line anchors and attention mechanisms to aggregate global information, while CLRNet [53] adopts learnable anchors and fuses multi-scale representations via a Feature Pyramid Network. Although these anchor-based strategies are generally faster and simpler than segmentation-based methods, they often struggle in highly varied or complex scenarios (e.g., extreme weather, occlusion, or sharply curved roads), where fixed anchors fail to adapt to lane shape variations.

2.2. Diffusion Models

Diffusion models [3, 6, 11, 15, 17, 18, 47] iteratively refine coarse inputs through a forward corruption process and a corresponding learned denoising procedure, enabling them to model complex data distributions with remarkable flexibility. While originally conceived for image generation, diffusion-based techniques have expanded to downstream tasks such as detection [6, 10], segmentation [2, 44], and other structured prediction problems [11, 47]. A large body of work adopts what is termed "hot diffusion," in which data are progressively corrupted by injecting Gaussian noise at each forward step [14, 38]. In this paradigm, the model learns to iteratively remove noise, moving from a random or partially disordered state back toward a clean sample.

However, in tasks with strongly constrained structures, hot diffusion can disrupt geometric priors by introducing random noise. Cold diffusion [3, 25, 49] addresses this by employing deterministic degradations, such as interpolation or masking, instead of stochastic noise, and then learning a restoration process. This avoids unrealistic perturbations and proves more suitable for semi-rigid configurations like molecular shapes. To our knowledge, no previous work has applied diffusion to lane detection, because hot diffusion, with its unconstrained noise injection, is ill-suited for such tasks. By starting from structured "anchors" rather than random noise, CDiffLane iteratively refines its estimates with fewer nonsensical deviations, improving both stability and efficiency in this highly constrained setting.

3. Proposed Approach

Throughout this paper, we denote *scalars* by italic lowercase letters (e.g., x), **vectors** by bold lowercase letters (e.g., $\mathbf{x}$), and **matrices** by bold uppercase letters (e.g., $\mathbf{X}$).

3.1. Preliminaries

Lane Representation. In 2D lane detection, as lanes typically appear in the lower part of an image, the input image is cropped before being annotated. A lane is then represented as a sequence of N (x, y) coordinates, denoted by $P = \{(x_1, y_1), \dots, (x_N, y_N)\}$, where the y coordinates of these points are uniformly distributed along the vertical axis of the cropped image, expressed as $y_i = \frac{H}{N-1} * i$ where H represents the height of the cropped image. Consequently, each x_i value corresponds to a specific y_i, establishing a clear mapping of lane points along the vertical axis. To reduce complexity, we parametrize each lane P by five parameters $(x_p, y_p, \theta_p, l, \delta_x)$ rather than predicting all $\{(x_i, y_i)\}$ coordinates directly. Here, (x_p, y_p) specifies the lane's starting position, θ_p is the initial slope, and l indicates the total lane length. The term δ_x captures incremental horizontal shifts at each sampled vertical step, which effectively modeling how the lane deviates from the naive straight line defined by the starting position, slope and length. This parameterization both simplifies learning and preserves sufficient flexibility to represent curved lane shapes.

Diffusion Model. While early diffusion methods typically rely on adding Gaussian noise in forward process [14, 36–38], recent work reveals that this step need not be noisy. In particular, cold diffusion [3] generalizes the notion of corruption to include any suitable degradation. Formally, rather than injecting noise, one defines a deterministic operator D that systematically degrades an initial sample $\mathbf{z}_0$. At each timestep $t \in \{1, 2, \dots, T\}$, the forward process is:

$$\mathbf{z}_t = q(\mathbf{z}_t | \mathbf{z}_0) = D(\mathbf{z}_0, t). \tag{1}$$

This flexibility allows diffusion models to better align with task-specific structural constraints without the restriction to noise injection. For instance, the corruption can be tailored such as blurring or interpolating. On the other hand, there exists a restoration operator R that inverts D, satisfying:

$$R(\mathbf{z}_t, t) = \mathbf{z}_0. \tag{2}$$

In real-world scenarios, R is typically implemented as a neural network with parameters θ. During training, R_θ takes the degraded latent sample $\mathbf{z}_t$ as input and is trained to predict the original sample $\mathbf{z}_0$. The training objective is commonly defined as the ℓ_2 loss between the model's prediction $\hat{\mathbf{z}_0}$ and $\mathbf{z}_0$:

$$\mathcal{L}_{obj} = \frac{1}{2} \| R_\theta(\mathbf{z}_t, t) - \mathbf{z}_0 \|^2. \tag{3}$$

At inference stage, $\mathbf{z}_0$ is reconstructed iteratively from the degraded sample $\mathbf{z}_t$ using the model R_θ, *i.e.*, $\mathbf{z}_T \to \mathbf{z}_{T-1} \to \cdots \to \mathbf{z}_0$. The update rule is as follows:

$$\mathbf{z}_{t-1} = \mathbf{z}_t - D(R_\theta(\mathbf{z}_t, t), t) + D(R_\theta(\mathbf{z}_t, t), t-1). \tag{4}$$

3.2. CDiffLane Architecture

Anchor-Based Diffusion Design. To address the challenges in lane detection, we propose a novel approach utilizing a diffusion model specifically designed to refine lane estimations progressively. Figure 2 illustrates the proposed CDiffLane framework. Unlike the approach in DiffusionDet [7], which applies Gaussian noise to degrade the ground-truth bounding box, our method aligns the ground-truth lane with predefined lane anchors. A lane anchor, represented as a straight line defined by parameters (x_p, y_p, θ_p), serves as a structured reference point to guide the detection process. In the lane diffusion process, we initiate with a collection of lane anchors $\mathbf{A} = \{\mathbf{a}_0, \mathbf{a}_1, \dots, \mathbf{a}_{N_s-1}\}$ where N_s represents the number of lane anchors. These anchors are initially distributed uniformly from left to right across the image, adhering to a slope that aligns with the general appearance of road lanes. Throughout the training process, these anchors dynamically update to fit the data distribution. The primary goal of this approach is to train a lane diffusion model R_θ that iteratively refines the model's predictions to converge on the target lane structure. The refinement process follows a series of diffusion steps T, formulated as follows:

$$\mathbf{A}^T \xrightarrow{R} \mathbf{A}^{T-1} \xrightarrow{R} \dots \xrightarrow{R} \mathbf{A}^0. \tag{5}$$

Our method integrates a continuous forward degradation process, leveraging a predefined anchor $\mathbf{a}$ and a ground-truth lane $\mathbf{a}_{gt}$. The degradation process is defined as:

$$\begin{aligned} \mathbf{a}^t &= D_\alpha(\mathbf{a}_{gt}, t) \\ &= \alpha_t \mathbf{a}_{gt} + (1 - \alpha_t)\mathbf{a}, \text{ with } \alpha_t \in [0, 1]. \end{aligned} \tag{6}$$

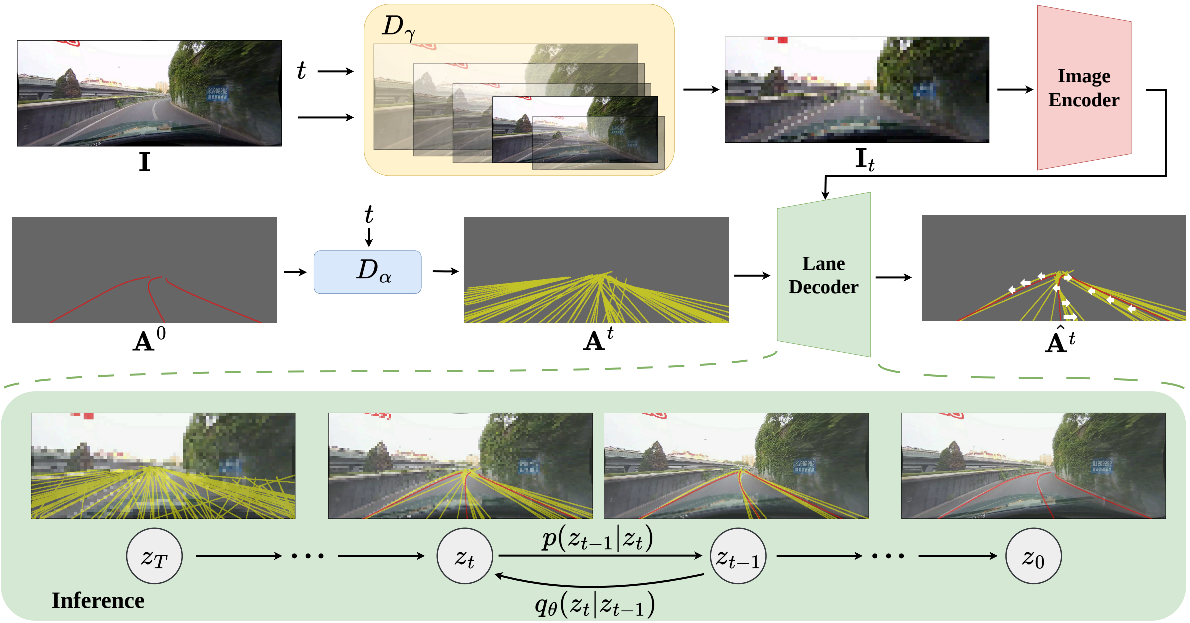

Figure 2. Flowchart of the proposed method. In the training phase, the model processes a degraded image and a lane degraded according to timestep t, based on the model's initial lane predictions, and outputs a refined version. During inference, the model begins with a predefined set of lane anchors and iteratively refines lane locations as the input image becomes progressively clearer.

Each anchor in $\mathbf{A}$ is matched to the closest ground-truth lane in the image. The ground-truth lane first extracts its (x, y, θ) components and duplicates them to match the number of anchors before the process begins. Here, $\mathbf{a}^t$ interpolates between the ground truth and the lane anchor, with α_t designed to decrease monotonically over time. This progressive shift ensures that the degraded lane representation transitions toward the anchor as time progresses. By employing this forward degradation strategy, the model initiates diffusion from an organized and semantically meaningful starting point, thereby avoiding the disordered noise that typically hampers performance.

In our iterative refinement process, our lane diffusion model R_θ is specifically designed to refine lane alignment and structure at each step t. Given a time step t and an image $\mathbf{I}$, we first rescale $\mathbf{I}$ to different resolutions based on t to obtain $\mathbf{I}_t$. Next, we input a degraded lane $\mathbf{A}^t$ along with the rescaled image $\mathbf{I}_t$ into R_θ to obtain the parameterized lane $\mathbf{A}^0$.[2] At each stage, the model provides the refined lane prediction as follows:

$$\Delta\mathbf{A}^t, \mathbf{l}^t, \delta^t, \mathbf{c}^t = R_\theta(\mathbf{A}^t, \mathbf{I}_t), \tag{7}$$

$$\hat{\mathbf{A}^0} = \mathbf{A}^t + \Delta\mathbf{A}^t, \tag{8}$$

where $\mathbf{c}^t$ represents the confidence score associated with the lane prediction $\mathbf{A}^t$ at time t. Notably, only the final step values of $\mathbf{l}^t, \delta^t$ and $\mathbf{c}^t$ are retained, while intermediate steps discard them. This iterative refinement enables the model to gradually enhance the lane estimation's accuracy and robustness, yielding high-confidence predictions that align closely with the true lane structure.

Resolution Scheduling and Learnable α. One of the key advantages of diffusion models in image generation is their ability to learn image structures in a progressive, coarse-to-fine manner. This gradual refinement enables the model to incrementally capture both low-level and high-level features, resulting in a highly detailed final output. However, in previous diffusion models for perception tasks [7, 34, 35, 45], the guidance for the diffusion process comes solely from a static conditioning input. When this conditioning input remains fixed throughout the process, the model tends to learn the same feature representations at each step. This limitation hinders the model's capacity to adaptively refine specific details in the image, as it continuously relies on unchanged information instead of dynamically adjusting to evolving input data.

To address this limitation, we propose a resolution scheduling mechanism for our lane detection model, introducing input images of varying resolution at each diffusion step. Specifically, we define a scheduling parameter γ_t, representing the factor by which the input image resolution is reduced at step t. Formally, the input image will be scaled down by a factor of γ_t and then scaled back, resulting in a lower quality effect. This parameter works with a resolution adjustment function, denoted D_γ, with the adjusted image at step t represented as $\mathbf{I}_t$. Intuitively, γ_t is designed to decrease gradually as t approaches zero, allowing the model

[2] The architecture of the restoration network can be found in the implementation details in Appendix.

to begin with blurred images and refine them progressively. This process allows the model to focus on high-level features in the early stages when t is large and to focus on finer, lower-level details as t decreases. This coarse-to-fine strategy encourages the model to incrementally improve its predictions, achieving a detailed and accurate final result.

Additionally, we introduce a learnable parameter α_t in Eq. 6, which determines the degradation ratio between the ground truth and a predefined anchor at each step. We interpret the difference in α values between consecutive steps as the variation magnitude v_t, defined as $v_t = \alpha_t - \alpha_{t-1}$. This variation magnitude is integrated into the objective function by weighting the loss at each timestep, allowing the model to adaptively adjust its learning process. For cases where the model struggles to generate precise predictions, a smaller variation magnitude reduces the impact of errors on the objective function, mitigating excessive corrections. In particular, when the input image resolution is too low to provide meaningful information, the model can minimize unnecessary modifications by dynamically reducing the magnitude. This adaptive mechanism enhances robustness across varying image qualities, improving the model to prioritize relevant features more effectively. Through this learnable adjustment, the model achieves a more context-aware refinement process, leading to improved accuracy and resilience in complex visual scenarios.

3.3. Training and Inference

Training Paradigm. Our training paradigm involves training a R_θ to predict the ground-truth $\mathbf{A}^0$ from predefined anchors. The pseudocode of the training procedure is provided in Alg. 1 in the Appendix. Specifically, given a degraded version of $\mathbf{A}^t$ derived from Eq. 6 and the corresponding $\mathbf{I}_t$ at timestep t, the goal is to reconstruct the ground-truth $\mathbf{A}^0$. A key challenge in this process is sampling drift in diffusion models—i.e., the misalignment between the training and sampling data distributions [5, 7, 17]. To mitigate this issue, following the approach in [17], we construct $\mathbf{A}^t$ based on the model's predictions rather than relying solely on the ground truth. More precisely, we first generate an initial estimate $\hat{\mathbf{A}}^0$ by feeding the anchor $\mathbf{A}^T$ and $\mathbf{I}_T$ into the detector. We then derive $\mathbf{A}^t$ by incorporating $\hat{\mathbf{A}}^0$ into Eq. 6, ensuring a more stable and consistent training process.

Training Loss. The output of the lane detector consists of N_s predicted lanes, each with an associated confidence score. During training, ground-truth lanes are dynamically assigned to one or more of the closest predicted lanes using the Hungarian algorithm. Following the approach outlined in [53], the overall training loss comprises a classification loss $\mathcal{L}_{cls}$ and a regression loss $\mathcal{L}_{reg}$, which account for lane existence and localization accuracy, respectively. Specifically, $\mathcal{L}_{cls}$ is a focal loss used for classification, as most predicted lanes remain unassigned and are labeled as 0. This design encourages the model to focus on assigned lanes, thereby enhancing prediction reliability. The regression loss $\mathcal{L}_{reg}$ consists of two components $\mathcal{L}_{xytl}$ and $\mathcal{L}_{iou}$. $\mathcal{L}_{xytl}$ is a smooth ℓ_1 applied to lane parameters (x_p, y_p, θ_p, l). $\mathcal{L}_{iou}$ is the lane IoU loss, computed over the sequence of lane points P. Lane points are first converted using the parameter set $(x_p, y_p, \theta_p, l, \delta_x)$ as described in Section 3.1. More details on lane IoU loss can be found in [53].

Additionally, we incorporate the variation magnitude v_t into the training loss function. During training, the regression loss is weighted by v_t, enabling the model to dynamically adjust the influence of regression at different timesteps. The total training loss is formulated as:

$$\begin{aligned}\mathcal{L}_{total} &= \lambda_{cls}\mathcal{L}_{cls} + v_t\mathcal{L}_{reg} \\ &= \lambda_{cls}\mathcal{L}_{cls} + v_t(\lambda_{xytl}\mathcal{L}_{xytl} + \mathcal{L}_{iou}),\end{aligned} \tag{9}$$

where λ_{cls} and λ_{xytl} are weighting coefficients. By applying the variation magnitude v_t to the lane regression loss, the model can dynamically modulate the impact of each step, improving its ability to handle the progressive nature of the lane refinement process.

Inference Procedure. The inference process in our method (Alg. 2 in the Appendix) adopts a diffusion-based sampling workflow designed to iteratively refine initial anchor-based estimations. We begin by placing lane anchors on a blurred version of the input image, ensuring that the model concentrates on overarching lane structures before fine details. At each subsequent timestep, clearer versions of the image are introduced—gradually increasing resolution or reducing blurring—while the lane anchors and offsets are updated to converge toward the true lane positions. This progressive refinement framework allows the model to incorporate high-level contextual cues from earlier, coarser steps and then focus on finer-grained corrections in later stages. As a result, anchor positions become more precise, enabling the model to consistently align with lane boundaries despite challenges like occlusion, road curvature, or adverse lighting conditions. Ultimately, the diffusion-based sampling procedure guides the model from an initial, anchor-driven guess to an accurate representation of real lane geometry.

4. Experiment

4.1. Experimental Setup

Datasets. We evaluate our method on two widely-recognized lane detection benchmarks, CULane [28] and TuSimple [1]. CULane features 88.9k training images, 9.7k validation images, and 34.7k test images (each 1640 × 590 pixels), spanning diverse environments such as urban streets, rural roads, and dense traffic. Its test set is categorized into nine conditions, e.g., crowded, night, and no line, to rigorously assess performance under varying real-world scenarios. In contrast, TuSimple is dedicated to highway settings, comprising 3.3k training images, 0.4k validation images, and 2.8k test images, all at 1280×720 resolution.

Evaluation metrics. Following previous works [39, 53], we assess lane prediction accuracy using the F1-measure for the CULane dataset, calculated via the Intersection-over-Union (IoU) between predicted and ground-truth lanes. In our experiments, we report $F_1@50$ and $F_1@75$, corresponding to IoU thresholds of 0.5 and 0.75, respectively. Notably, the higher threshold (0.75) serves as a more rigorous test for precise lane localization, emphasizing improvements in fine-grained accuracy. Additionally, we include the mF_1 score, which represents the average score over $F_1@50$ to $F_1@95$. For the TuSimple dataset, a predicted lane is considered correct if more than 85% of its pixels overlap with those in the ground truth. Due to space constraints, details are provided in the Appendix.

Implementation Details. We build our restoration network R_θ on CLRNet [53], using a backbone (e.g., ResNet [13] or DLA [50]) plus an FPN [21] to extract multi-scale features. These features are then cropped according to the degraded lane $\mathbf{A}^t$ before generating final lane predictions and confidence scores. Due to space constraints, full implementation details including training configurations, data preprocessing, and inference procedures are provided in the Appendix.

4.2. Quantitative Results

Performance on CULane. As shown in Table 1, our *CDiffLane* outperforms the state-of-the-art methods on the CULane test set, achieving an F_1 score of **81.26** at 50% IoU ($F_1@50$) and **65.31** at 75% IoU ($F_1@75$). Compared to the baseline (CLRNet with a DLA-34 backbone), we obtain a 0.79-point gain on $F_1@50$ and a 2.53-point improvement on $F_1@75$. While our result shows a modest 0.13-point improvement over GSENet at $F_1@50$, the margin increases to 1.23 points at $F_1@75$, highlighting stronger localization precision. The smaller gain at $F_1@50$ aligns with previous oracle experiments [53], which found that most lanes were already detected correctly, though misclassification remained a challenge. As our cold diffusion framework focuses on refining geometric placement rather than classification, improvements at this lower threshold are limited. However, the stricter $F_1@75$ threshold demands more precise lane alignment, where iterative refinement proves effective. By progressively correcting misaligned anchors, CDiffLane significantly enhances performance at higher IoU thresholds.

Moreover, Table 1 also manifest that our method reduces false positives (FP) and demonstrates robust performance in cross-scenario tests (e.g., "night" or "crowded" conditions). We attribute this to the iterative nature of our diffusion process, which converges to plausible lane boundaries over multiple steps, rather than relying on a single-pass estimate. These findings highlight two key advantages of employing cold diffusion with anchor priors. First, the iterative design refines uncertain anchor positions step by step, reducing FP rates. Second, by using deterministic degradations instead of random noise, we preserve structural consistency for lanes under challenging conditions (e.g., severe occlusions or tight curves). Hence, the cold diffusion approach offers notable benefits in aligning lane detection models with stricter precision requirements.

Performance on TuSimple. Table 2 compares our approach against existing methods on the TuSimple benchmark. While the slight performance gaps suggest an overall trend toward saturation on this dataset, *CDiffLane* still achieves a notably high $F_1@50$ of **97.71**. In addition, our method reaches a **1.62%** false positive rate (FP), improving upon previous scores by 0.39 points. These results indicate that our framework is capable of pushing the limits of accuracy in highway-specific lane detection tasks, even as top-line metrics converge.

Despite a somewhat higher false negative rate (FN) compared with certain competitors, it is important to recognize the trade-off inherent in lane detection: methods targeting a lower FN often forecast more lanes to ensure coverage, thus inflating FP. In contrast, *CDiffLane* maintains a more balanced strategy, prioritizing minimal spurious lane predictions (FP) while retaining competitive FN performance. This trade-off proves advantageous in practice, as high FP can degrade driving decisions by incorrectly signaling additional lanes, whereas our method focuses on achieving strong localization with fewer extraneous detections. Consequently, the overall robustness of *CDiffLane* ensures that missed lanes are kept at acceptable levels, while false positives are substantially reduced, leading to a favorable balance for real-world autonomous driving scenarios.

4.3. Qualitative Results

Figure 3 demonstrates example outputs on the CULane test set under varied and difficult conditions, such as low visibility, shadows, and pronounced road curvature. Traditional anchor-based approaches (e.g., LaneATT, CLRNet) frequently produce false positives (highlighted in red), as they rely on *one-shot* anchor initialization. This single-pass strategy struggles to account for large contextual shifts or unusual lane shapes, leading to higher error rates when lanes deviate from predefined templates. In contrast, our method adopts a multi-step refinement process, iteratively refining lane predictions as it receives progressively more detailed image cues. This feedback loop not only captures richer global context but also compensates for suboptimal anchor initialization. Consequently, false positives diminish, and overall accuracy rises, demonstrating that an iterative diffusion approach can better handle diverse road layouts and environmental conditions, even in complex scenarios like nighttime driving or curved lane segments.

Method	Backbone	Venue	$mF_1\uparrow$	F_1@50$\uparrow$	F_1@75$\uparrow$	Normal↑	Crowd↑	Dazzle↑	Shadow↑	Noline↑	Arrow↑	Curve↑	Cross↓	Night↑
CondLane [22]	ResNet-101	ICCV'21	54.83	79.48	61.23	93.47	77.44	70.93	80.91	54.13	90.16	75.21	1201	74.80
GANet [42]	ResNet-101	CVPR'22	-	79.63	-	93.67	78.66	71.82	78.32	53.38	89.86	**77.37**	1352	73.85
CLRNet [53]	ResNet-101	CVPR'22	55.55	80.13	62.96	93.85	78.78	72.49	82.33	54.50	89.79	75.57	1262	75.51
CLRNet [53]	DLA-34	CVPR'22	55.64	80.47	62.78	93.73	79.59	75.30	82.51	54.58	90.62	74.13	1155	75.37
CondLSTR [8]	ResNet-101	ICCV'23	-	80.77	-	**94.17**	79.90	75.43	80.99	55.00	**90.97**	73.83	1047	75.11
CLRerNet [16]	ResNet-101	WACV'24	-	80.91	64.30	93.91	80.03	72.98	82.92	55.73	90.53	74.67	1113	76.13
CLRerNet [16]	DLA-34	WACV'24	-	81.12	64.07	94.02	80.20	74.41	83.71	**56.27**	90.39	74.67	1161	76.53
GSENet [39]	ResNet-101	AAAI'24	56.53	80.84	64.23	94.05	79.90	74.94	82.21	55.63	90.78	-	1164	76.08
GSENet [39]	DLA-34	AAAI'24	56.45	81.13	64.08	93.91	**80.30**	**76.36**	83.41	56.25	90.36	-	1036	76.26
Lane2Seq(anchor) [54]	ViT-Base	CVPR'24	-	79.27	-	93.11	77.43	73.25	79.46	53.74	90.02	72.44	1173	75.12
CDiffLane	DLA-34	-	**58.06**	**81.26**	**65.31**	93.93	80.26	72.64	**84.04**	55.20	90.85	74.86	**893**	**76.60**

Table 1. Comparison of leading lane detection models on the CULane test set. All scenarios use the F_1 score at 0.5 IoU, except *Cross*, which reports only false positives. **Bold** highlights the best result, and underlines mark the second-best.

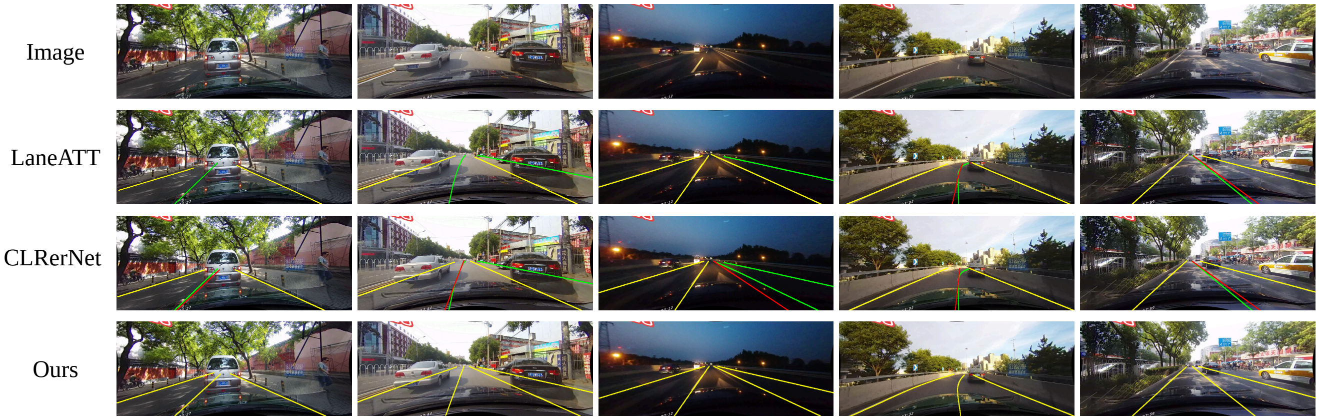

Figure 3. Visualization results of LaneATT, CLRerNet, and our method on the CULane. Yellow lines represent correctly predicted lanes (IoU above the threshold), classified as True Positives (TP). Red lines indicate mispredicted lanes (IoU below the threshold), categorized as False Positives (FP). Green lines denote False Negatives (FN), representing lanes present in the ground truth but missed by the model.

Method	Backbone	$F_1\uparrow$	Acc↑	FP↓	FN↓
SCNN [30]	VGG16	95.97	96.53	6.17	**1.80**
RESA [52]	ResNet-34	96.93	**96.82**	3.63	2.48
LaneATT [40]	ResNet-122	96.06	96.10	5.64	2.17
CondLane [22]	ResNet-101	97.24	96.54	2.01	3.50
CANet [48]	ResNet-34	97.44	96.66	2.32	2.79
CDiffLane	ResNet-34	**97.71**	96.67	**1.62**	3.01

Table 2. Quantitative Result on TuSimple.

Diffusion	Image Degrad.	F_1@50	F_1@75
Hot	None	79.25	60.77
Cold	None	**80.88**	**64.56**
Cold	Resolution	81.26	**65.31**
Cold	DCT	**81.29**	65.17
Cold	Gaussian Blur	81.06	64.78
Cold	Noise	80.93	64.52

Table 3. Ablation study on image degradation and diffusion version in CULane dataset.

4.4. Ablation study

To investigate how each component of our framework contributes to the final performance, we conduct a series of ablation experiments on the CULane dataset. The F_1 scores at 50% and 75% IoU thresholds are reported in Table 3, offering a clear comparison of different diffusion paradigms (hot vs. cold) and various image degradation strategies.

Hot vs. Cold Diffusion. We first evaluate the commonly used "hot diffusion", which adds Gaussian noise to lane anchors and initiates the denoising process from a Gaussian distribution. While this technique has shown effectiveness in object detection or segmentation [2, 7], it proves suboptimal for lane detection: it yields F_1@50/F_1@75 of $79.25/60.77$, underperforming even the baseline CLRNet. We attribute this to hot diffusion's random noise initialization, which often places lane anchors in unrealistic regions. By contrast, our "cold diffusion" begins with a structured lane anchor that better reflects plausible lane locations, thereby reducing extraneous anchor positions. As a result, switching to cold diffusion alone raises performance to 80.88/64.56 (F_1@50/F_1@75), highlighting the value of a deterministic, anchor-driven starting point.

Sche. α	Sche. γ	Lrn. α	Lrn. γ	F_1@50	F_1@75
✓	✓			81.04	65.11
✓			✓	80.93	64.95
	✓	✓		**81.26**	**65.31**
		✓	✓	81.09	65.03

Table 4. Ablation study on α and γ. 'Sche.' abbreviates schedule, and 'Lrn.' abbreviates learnable. The schedule α follows a cosine scheduling, while γ. uses a linear schedule.

Image Degradation Strategies. Next, we examine how various image degradation operations affect the coarse-to-fine refinement process. Beyond using no explicit image degradation ("None"), we test:

1. *Resolution*: Downsampling and then upsampling each image to progressively introduce clarity,
2. *DCT*: Filtering high-frequency components via discrete cosine transform before reapplying them in later steps,
3. *Gaussian Blur*: Convolving the image to diffuse edges and reduce sharp details,
4. *Noise*: Injecting random noise at each step.

Our results show that *Resolution* and *DCT* yield the largest gains among all degradations (see Table 3). The former achieves $81.26/65.31$, while the latter reaches $81.29/65.17$ at F_1@50/F_1@75. Both techniques prioritize the reconstruction of coarse shapes and progressively refine finer details over multiple iterations. This approach aligns well with cold diffusion's anchor-based iterative refinement strategy, ensuring a structured and coherent recovery process. In contrast, *Gaussian Blur* demonstrates notably inferior performance despite its superficial resemblance to *Resolution*. We attribute this disparity to the critical role of lane boundaries in lane detection. While *Resolution* maintains these essential edge features, *Gaussian Blur* compromises them by smearing and weakening the defining contours, thereby reducing the model's capacity to accurately identify lane markings. Similarly, the introduction of *Noise* at each iteration disrupts the localized, anchor-centric design inherent in CLRNet, further exacerbating structural degradation and leading to suboptimal F_1 scores. Overall, our findings highlight the necessity of preserving lane geometry and contextual integrity in the diffusion process. They further validate that the careful selection of image degradation operations can significantly enhance the accuracy of cold diffusion-based lane detection. These insights underscore the importance of tailored preprocessing strategies to optimize model performance in real-world applications.

Learnable α vs. Scheduled γ. We further investigate the effect of treating the parameters α (lane-anchor interpolation) and γ (image-resolution factor) as either *learnable* or *fixed/scheduled*. Our findings show that the best performance arises when α is learnable and γ is maintained on a fixed schedule, giving $81.26/65.31$ for $F_1@50/F_1@75$. A learnable α dynamically adjusts the weighting of each iterative refinement step, amplifying confident predictions while minimizing the influence of uncertain ones. In contrast, making γ learnable disrupts the model's training consistency; the image input then shifts unpredictably at each step, scattering the model's attention and degrading fine-grained alignment. Consequently, scheduling γ in a stable, predetermined way emerges as the more effective strategy, balancing coherent refinement with the flexibility needed for high-precision lane detection.We perform an experiment to examine how mF_1 varies with the number of inference steps and the resulting speed (FPS). As shown in Fig. 4, the iterative strategy indeed improves performance but lowers the number of tasks the model can process per second. While this trade-off may be acceptable for tasks such as high-density map construction, it can be less ideal in time-critical settings. Notably, even when reducing the iterative steps to just one—yielding an FPS above 140—our CDiffLane still surpasses GSENet, highlighting its robustness under lower iteration budgets.

4.5. Limitation

While *CDiffLane* improves detection accuracy through iterative refinement, the diffusion process naturally adds extra inference steps. One promising approach to alleviate this overhead is *progressive distillation*, where a student model inherits the diffusion model's knowledge and produces reliable predictions in fewer iterations, thus preserving accuracy while lowering computational cost. As the first to integrate diffusion into lane detection, our main objective has been to raise accuracy; future efforts will focus on optimizing for the real-time scenarios. Notably, even at a single iterative step (exceeding 140 FPS), *CDiffLane* still surpasses GSENet, demonstrating strong performance under reduced iteration budgets (see Appendix C for details).

5. Conclusion

In this work, we propose *CDiffLane*, which integrates the cold diffusion framework into anchor-based lane detection. To enhance the model's coarse-to-fine prediction, we introduce resolution scheduling and a learnable α , which provide sequential information that enables the model to iteratively refine lane predictions with greater precision. Extensive experiments demonstrate that CDiffLane outperforms state-of-the-art baselines in accuracy, establishing its effectiveness in lane detection tasks. As the first framework to integrate cold diffusion into a visual perception task, we anticipate that this architecture can be extended to other perception-based applications, further advancing the field of computer vision.

Acknowledgments

This work is partially supported by the National Science and Technology Council (NSTC), Taiwan (Grants: NSTC-112-2221-E-A49-059-MY3, NSTC-112-2221-E-A49-094-MY3, NSTC-112-2628-E-002-033-MY4), and by the Center of Data Intelligence, National Taiwan University (Grants: 114L900901, 114L900902, 114L900903), through the Higher Education Sprout Project, MOE, Taiwan. Additional support is provided by the Industry-Academia Innovation School, NYCU (Project No. 113UA10024-2), under the MOE's National Key Fields Program, and the Hon Hai Research Institute.

References

[1] Tusimple benchmark. https://github.com/TuSimple/tusimple-benchmark/. 2, 5

[2] Tomer Amit, Tal Shaharbany, Eliya Nachmani, and Lior Wolf. Segdiff: Image segmentation with diffusion probabilistic models. *arXiv preprint arXiv:2112.00390*, 2021. 2, 7

[3] Arpit Bansal, Eitan Borgnia, Hong-Min Chu, Jie Li, Hamid Kazemi, Furong Huang, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Cold diffusion: Inverting arbitrary image transforms without noise. *NeurIPS*, 2024. 2, 3

[4] Qing Chang and Yifei Tong. A hybrid global-local perception network for lane detection. In *AAAI*, 2024. 1

[5] Sitan Chen, Sinho Chewi, Jerry Li, Yuanzhi Li, Adil Salim, and Anru Zhang. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. In *ICLR*, 2023. 5

[6] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. Diffusiondet: Diffusion model for object detection. In *ICCV*, 2023. 1, 2

[7] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. Diffusiondet: Diffusion model for object detection. In *ICCV*, 2023. 3, 4, 5, 7

[8] Ziye Chen, Yu Liu, Mingming Gong, Bo Du, Guoqi Qian, and Kate Smith-Miles. Generating dynamic kernels via transformers for lane detection. In *ICCV*, 2023. 7

[9] Ruochen Fan, Xuanrun Wang, Qibin Hou, Hanchao Liu, and Tai-Jiang Mu. Spinnet: Spinning convolutional network for lane boundary detection. *Computational Visual Media*, 2019. 2

[10] Haoyang Fang, Boran Han, Shuai Zhang, Su Zhou, Cuixiong Hu, and Wen-Ming Ye. Data augmentation for object detection via controllable diffusion models. In *WACV*, 2024. 2

[11] Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and Lingpeng Kong. DiffuSeq: Sequence to sequence text generation with diffusion models. In *ICLR*, 2023. 2

[12] Jie Guo, Zhihua Wei, and Duoqian Miao. Lane detection method based on improved ransac algorithm. In *IEEE Twelfth International Symposium on Autonomous Decentralized Systems*, 2015. 1

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6, 1

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 2020. 2, 3

[15] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *Journal of Machine Learning Research*, 2022. 2

[16] Hiroto Honda and Yusuke Uchida. Clrernet: improving confidence of lane detection with laneiou. In *WACV*, 2024. 1, 7

[17] Yuanfeng Ji, Zhe Chen, Enze Xie, Lanqing Hong, Xihui Liu, Zhaoqiang Liu, Tong Lu, Zhenguo Li, and Ping Luo. Ddp: Diffusion model for dense visual prediction. In *ICCV*, 2023. 2, 5

[18] Lingkai Kong, Jiaming Cui, Haotian Sun, Yuchen Zhuang, B Aditya Prakash, and Chao Zhang. Autoregressive diffusion model for graph generation. In *ICML*, 2023. 2

[19] Xiang Li, Jun Li, Xiaolin Hu, and Jian Yang. Line-cnn: End-to-end traffic line detection with line proposal unit. *IEEE Transactions on Intelligent Transportation Systems*, 2019. 1, 2

[20] Zuo-Quan Li, Hui-Min Ma, and Zheng-Yu Liu. Road lane detection with gabor filters. In *ISAI*, 2016. 1

[21] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 6, 1

[22] Lizhe Liu, Xiaohao Chen, Siyu Zhu, and Ping Tan. Condlanenet: a top-to-down lane detection framework based on conditional convolution. In *ICCV*, 2021. 2, 7

[23] Ruijin Liu, Zejian Yuan, Tie Liu, and Zhiliang Xiong. End-to-end lane shape prediction with transformers. In *WACV*, 2021. 1, 2

[24] Chunyang Mu and Xing Ma. Lane detection based on object segmentation and piecewise fitting. *TELKOMNIKA Indonesian Journal of Electrical Engineering*, 2014. 1

[25] Sergio Naval Marimont, Vasilis Siomos, Matthew Baugh, Christos Tzelepis, Bernhard Kainz, and Giacomo Tarroni. Ensembled cold-diffusion restorations for unsupervised anomaly detection. In *MICCAI*, 2024. 3

[26] Davy Neven, Bert De Brabandere, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Towards end-to-end lane detection: an instance segmentation approach. In *IEEE intelligent vehicles symposium (IV)*, 2018. 1

[27] Jianwei Niu, Jie Lu, Mingliang Xu, Pei Lv, and Xiaoke Zhao. Robust lane detection using two-stage feature extraction with curve fitting. *PR*, 2016. 1

[28] Xingang Pan, Jianping Shi, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Spatial as deep: Spatial cnn for traffic scene understanding. In *AAAI*, 2018. 1, 2, 5

[29] Xingang Pan, Jianping Shi, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Spatial as deep: Spatial cnn for traffic scene understanding. In *AAAI*, 2018. 2

[30] Angshuman Parashar, Minsoo Rhu, Anurag Mukkara, Antonio Puglielli, Rangharajan Venkatesan, Brucek Khailany, Joel Emer, Stephen W Keckler, and William J Dally. Scnn:

An accelerator for compressed-sparse convolutional neural networks. *ACM SIGARCH computer architecture news*, 2017. 7

[31] Jonah Philion. Fastdraw: Addressing the long tail of lane detection by adapting a sequential prediction network. In *CVPR*, 2019. 1

[32] Zequn Qin, Huanyu Wang, and Xi Li. Ultra fast structure-aware deep lane detection. In *ECCV*, 2020. 1, 2

[33] Zequn Qin, Pengyi Zhang, and Xi Li. Ultra fast deep lane detection with hybrid anchor driven ordinal classification. *IEEE transactions on pattern analysis and machine intelligence*, 2022. 1

[34] Yasiru Ranasinghe, Nithin Gopalakrishnan Nair, Wele Gedara Chaminda Bandara, and Vishal M Patel. Crowddiff: Multi-hypothesis crowd density estimation using diffusion models. In *CVPR*, 2024. 4

[35] Saurabh Saxena, Charles Herrmann, Junhwa Hur, Abhishek Kar, Mohammad Norouzi, Deqing Sun, and David J Fleet. The surprising effectiveness of diffusion models for optical flow and monocular depth estimation. *CVPR*, 2024. 4

[36] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015. 3

[37] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021.

[38] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *NeurIPS*, 2019. 2, 3

[39] Junhao Su, Zhenghan Chen, Chenghao He, Dongzhi Guan, Changpeng Cai, Tongxi Zhou, Jiashen Wei, Wenhua Tian, and Zhihuai Xie. Gsenet: Global semantic enhancement network for lane detection. In *AAAI*, 2024. 1, 6, 7

[40] Lucas Tabelini, Rodrigo Berriel, Thiago M Paixao, Claudine Badue, Alberto F De Souza, and Thiago Oliveira-Santos. Keep your eyes on the lane: Real-time attention-guided lane detection. In *CVPR*, 2021. 1, 2, 7

[41] Lucas Tabelini, Rodrigo Berriel, Thiago M Paixao, Claudine Badue, Alberto F De Souza, and Thiago Oliveira-Santos. Polylanenet: Lane estimation via deep polynomial regression. In *ICPR*, 2021. 1, 2

[42] Jinsheng Wang, Yinchao Ma, Shaofei Huang, Tianrui Hui, Fei Wang, Chen Qian, and Tianzhu Zhang. A keypoint-based global association network for lane detection. In *CVPR*, 2022. 7

[43] Chunyu Wei, Hailong Li, Junyi Shi, Guoyang Zhao, Huaiqu Feng, and Longzhe Quan. Row anchor selection classification method for early-stage crop row-following. *Computers and Electronics in Agriculture*, 2022. 1

[44] Junde Wu, Rao Fu, Huihui Fang, Yu Zhang, Yehui Yang, Haoyi Xiong, Huiying Liu, and Yanwu Xu. Medsegdiff: Medical image segmentation with diffusion probabilistic model. In *MIDL*, 2024. 2

[45] Fei Xie, Zhongdao Wang, and Chao Ma. Diffusiontrack: Point set diffusion model for visual object tracking. In *CVPR*, 2024. 4

[46] Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. Geodiff: A geometric diffusion model for molecular conformation generation. In *ICLR*, 2022. 2

[47] Dongchao Yang, Jianwei Yu, Helin Wang, Wen Wang, Chao Weng, Yuexian Zou, and Dong Yu. Diffsound: Discrete diffusion model for text-to-sound generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023. 2

[48] Zhongyu Yang, Chen Shen, Wei Shao, Tengfei Xing, Runbo Hu, Pengfei Xu, Hua Chai, and Ruini Xue. Canet: Curved guide line network with adaptive decoder for lane detection. In *ICASSP*, 2023. 7

[49] Hao Yen, François G Germain, Gordon Wichern, and Jonathan Le Roux. Cold diffusion for speech enhancement. In *ICASSP*, 2023. 3

[50] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *CVPR*, 2018. 6, 1

[51] Jia-Qi Zhang, Hao-Bin Duan, Jun-Long Chen, Ariel Shamir, and Miao Wang. Houghlanenet: Lane detection with deep hough transform and dynamic convolution. *Computers & Graphics*, 2023. 2

[52] Tu Zheng, Hao Fang, Yi Zhang, Wenjian Tang, Zheng Yang, Haifeng Liu, and Deng Cai. Resa: Recurrent feature-shift aggregator for lane detection. In *AAAI*, 2021. 2, 7

[53] Tu Zheng, Yifei Huang, Yang Liu, Wenjian Tang, Zheng Yang, Deng Cai, and Xiaofei He. Clrnet: Cross layer refinement network for lane detection. In *CVPR*, 2022. 1, 2, 5, 6, 7

[54] Kunyang Zhou. Lane2seq: towards unified lane detection via sequence generation. In *CVPR*, 2024. 7