SIGMA-Lane: Scale-pyramId Gated MAmba for Temporally Consistent Video Lane Detection

Tiancheng Zhang, Yan Gao, Xiangjie Kong, Guojiang Shen, Jiaxin Du, and Mengmeng Wang†

Zhejiang Key Laboratory of Visual Information Intelligent Processing, College of Computer Science and Technology, Zhejiang University of Technology

Abstract. Video lane detection requires predictions that remain stable across frames, yet severe vehicle occlusions can break temporal cues. In streaming recurrent models, corrupted observations may enter the hidden state and produce errors that persist into later frames. Existing occlusion-aware refinements usually provide obstacle masks as auxiliary inputs, so the state-update path is only indirectly protected. We propose SIGMA-Lane, which treats this failure mode as *state contamination* in State Space Model (SSM)-based temporal modeling. SIGMA-Lane places occlusion-aware gates on the SSM write and residual-fusion paths, controlling how current observations enter temporal memory and are fused back after temporal propagation. After coordinate-consistent affine alignment, the model combines two complementary paths: SSM-consistent dual-gating for temporal filtering and Structural Spatial Retrieval (SSR) for recovering missing lane structure from aligned historical priors. Experiments on VIL-100 and OpenLane-V show improved temporal stability under heavy occlusion, with competitive F1 and mIoU scores.

Keywords: Video Lane Detection · State Space Models · Temporal Consistency · Occlusion Handling

1 Introduction

Lane detection provides structured road-boundary information for downstream planning and control in autonomous driving. Existing methods achieve strong per-frame accuracy with segmentation, parametric, or anchor-based representations [19, 23, 37]. However, in video, the detector must also keep the predictions consistent over time. Two temporal designs are common. Window-batched methods [27, 28, 35, 38] jointly process a temporal window (Fig. 1a). Recurrent methods [2, 12, 34, 41] propagate a state frame by frame for streaming inference (Fig. 1b). Both designs are vulnerable when large vehicles occlude lanes for several frames. If corrupted observations enter the temporal pipeline, they can affect later predictions unless the update itself suppresses them.

Existing recursive methods, such as OMR [11], feed obstacle masks into a ConvLSTM-based refinement module. This gives the model obstacle context, but

† Corresponding author:Mengmeng Wang.

it does not directly constrain how occluded features are written into the recurrent state. We examine this issue through the SSM update $h_t = \bar{A}_t h_{t-1} + \bar{B}_t x_t$, where the current observation enters the state through the write term. Mask-based noise filtering is common in vision, but in an SSM, this write term can still inject occlusion-corrupted features into the hidden state and leave residual errors across frames. Once stored, the corrupted feature becomes part of temporal memory and is reused by later predictions. We call this failure mode *state contamination*. This perspective motivates placing the obstacle prior on the SSM write path and the residual-fusion path, instead of using it only as an auxiliary input. It also exposes a second issue, *coordinate inconsistency*: if features and masks are warped independently, the gate may no longer match the occluded region it is meant to modulate.

We propose SIGMA-Lane (Fig. 1c), an SSM-based framework that gates the state update under occlusion. SSM-Consistent Dual-Gating reduces the effective write amplitude of $\bar{B}_t x_t$ at the input stage and controls residual fusion after temporal propagation. Lane-Guided Affine Warp applies the same affine transform to features, obstacle masks, and lane masks, keeping the gating signal in the same coordinate frame as the feature being modulated. SSR then recovers lane structure from aligned historical features with mask-guided cross-attention. A geometry-aware start token initializes the temporal state at the beginning of a stream. On VIL-100 [35], SIGMA-Lane achieves the best mIoU, F1@0.5, accuracy, and temporal stability scores, reducing R_F/R_M@0.5 to 0.023/0.030. On OpenLane-V [12], it obtains the best F1@0.5/F1@0.8 and the lowest persistent missing rate at the stricter threshold (R_M@0.8 = 0.384), while remaining competitive in mIoU (Tab. 1, Tab. 2).

Our contributions are:

- We formulate state contamination as a failure mode in SSM-based video lane detection, and introduce a dual-gating mechanism that controls both the SSM write path and residual-fusion path under occlusion.
- We design a dual-path aggregation module with coordinate-consistent affine alignment, Mamba-based temporal propagation, and Structural Spatial Retrieval. The design separates temporal filtering from spatial recovery, addressing temporal-state contamination and loss of lane topology with different modules.
- On VIL-100 and OpenLane-V, SIGMA-Lane obtains competitive accuracy with lower flickering and missing rates. Ablations and controlled contamination analysis show that dual-gating accounts for a large share of the improvement.

2 Related Work

Lane Detection Representations. Modern lane detectors use several representation strategies for complex road scenes. Segmentation-based methods [9,10, 19–21,36] treat lane detection as pixel-wise classification and improve spatial coherence with message passing [19] or multi-scale aggregation [21,36]. Parametric

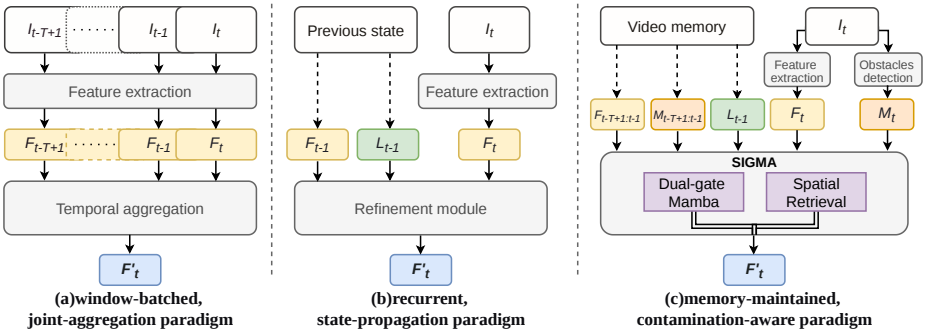

Fig. 1: Comparison of Video Lane Detection Paradigms. The window-batched, joint-aggregation paradigm (a) extracts features $F_{t-T+1:t}$ from a fixed temporal window and aggregates them jointly, incurring high latency without reusing past computation. The recurrent, state-propagation paradigm (b) propagates the previous state (F_{t-1}, L_{t-1}) frame by frame, enabling streaming inference, but offers no mechanism to prevent corrupted observations from contaminating the hidden state under occlusion. Our memory-maintained, contamination-aware paradigm (c) maintains a video memory bank of historical features F, obstacle masks M, and lane masks L. After lane-guided affine alignment, two paths process the aligned data: SSM-Consistent Dual-Gating for occlusion-aware temporal filtering and SSR for lane-guided spatial recovery.

methods [5, 16, 18, 24, 26, 31, 32, 39] model lanes as polynomials or Bézier curves; CondLaneNet [15], for example, uses conditional convolution for instance-level prediction. Anchor-based detectors [3, 23, 29, 37] use predefined line priors for localization. These single-frame methods perform well on static benchmarks, but they process each image independently and cannot use temporal context to infer lanes hidden by occlusion.

Temporal Context in Video Lane Detection. Extending detection to videos reduces the instability of single-frame methods. Early recurrent designs [34, 41] use ConvLSTM or ConvGRU to model frame-to-frame dependencies. Recent transformers [25], including LaneTCA [38] and MMA-Net [35], use attention for global temporal context. Existing temporal aggregators still face a trade-off: recursive methods such as RVLD [12] and OMR [11] are efficient but can propagate corrupted observations, while attention-based models are more expensive. Current efficient solutions often use auxiliary motion correction [12] or separate memory refinement [11], but they do not control how occluded evidence enters the temporal state. SIGMA-Lane targets this gap through a contamination-aware update rule.

SSMs for Sequence Modeling. SSMs [6, 7] model long sequences with linear complexity. They have been used in generic vision tasks [14, 17, 40] and autonomous driving tasks, including 3D perception [33], motion planning [22], and static lane detection [1]. Mamba adds selective scanning for content-aware sequence modeling. In occluded lane videos, however, contaminated observations may still enter the hidden state and accumulate over time. Our work com-

plements Mamba's internal selectivity with an obstacle prior at the write and residual-fusion paths.

3 Method

3.1 Preliminary

We build on OMR's recursive video lane detection framework [11]. We retain its encoder and decoder but replace its aggregation module. Given a current frame I_t, a ResNet18 [8] encoder extracts a multi-scale feature map $F_t \in \mathbb{R}^{B\times C\times H\times W}$. The aggregation module refines F_t into enhanced features $\tilde{F}_t$ using temporal-state propagation and spatial retrieval. Following the eigenlane paradigm [13], the decoder predicts a probability map P_t and a coefficient map C_t. Lane instances are then extracted by NMS to obtain a binary lane mask L_t.

For streaming inference, a video memory bank stores the past T frames' aggregated features and obstacle masks. With spatial positions flattened, these tensors are $\mathbf{F} \in \mathbb{R}^{(BHW)\times T\times C}$ and $\mathbf{M} \in \mathbb{R}^{(BHW)\times T\times 1}$. The bank also stores the previous lane mask L_{t-1} for alignment and guidance. At the end of each frame, $\tilde{F}_t$, M_t, and L_t are written back to memory. A geometry-aware start token initializes this recursive process at a cold start (Sect. 3.5). OMR uses ConvLSTM for temporal refinement; we use Mamba2 [4] as the temporal update core and modify the aggregation module to address state contamination and coordinate inconsistency. SSM-Consistent Dual-Gating is the central temporal filter. Lane-guided alignment supplies coordinate-consistent masks, and SSR retrieves missing lane structure after gated temporal filtering. Fig. 2 shows the overall architecture.

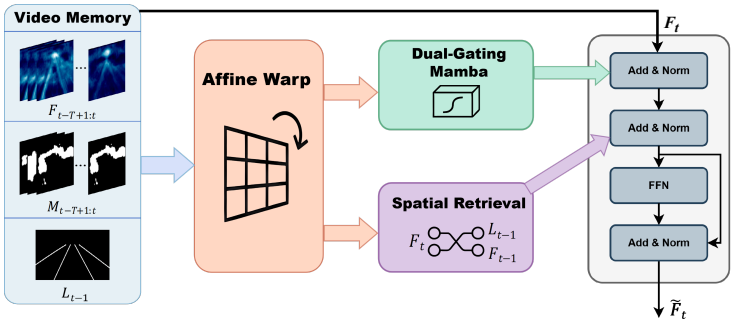

Fig. 2: Overview of the SIGMA-Lane Aggregation Module. Video Memory stores historical features $F_{t-T+1:t}$, obstacle masks $M_{t-T+1:t}$, and the previous lane mask L_{t-1}. The Affine Warp module aligns historical data to current-frame coordinates. Warped features then pass through two paths: (1) SSM-Consistent Dual-Gating with a scale-pyramid design for multi-scale temporal propagation and occlusion-aware gating, (2) SSR where current features F_t retrieve spatial context from warped F_{t-1} enhanced by L_{t-1} embeddings. The outputs are aggregated with current-frame features F_t via Add&Norm and FFN blocks, producing enhanced features $\tilde{F}_t$.

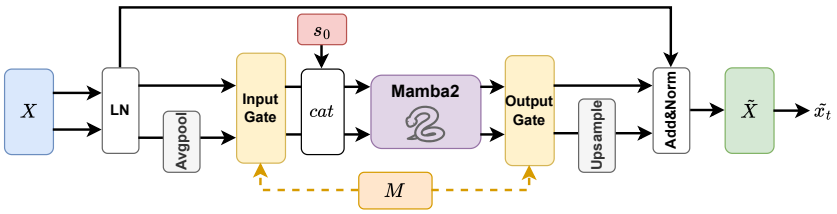

Fig. 3: SSM-Consistent Dual-Gating with a scale-pyramid design. Input features X pass through LN and are split into high- and low-resolution branches. Each branch applies Input Gate, concatenates the Geometry-Aware Start Token s_0, performs Mamba2 [4] temporal modeling, and applies Output Gate, with both gates modulated by the obstacle mask M. The low-resolution branch is upsampled and merged with the high-resolution branch through Add&Norm to produce $\tilde{X}$. Finally, the current-frame features $\tilde{x}_t$ are extracted from $\tilde{X}$ and sent to subsequent modules.

3.2 SSM-Consistent Dual-Gating

Observations from occluded regions are noisy. Directly writing them into the temporal state causes *state contamination* that accumulates through recursive propagation. Starting from the linear update rule of SSMs,

$$h_t = \bar{A}_t h_{t-1} + \bar{B}_t x_t, \tag{1}$$

the central design question is where the obstacle prior enters the recursion. We gate two pathways where occlusion noise can enter the temporal output: the SSM write path, which controls what enters the temporal state, and the residual-fusion path, which controls how much current-frame evidence is mixed back after temporal propagation. The input gate acts before selective scanning and modulates the SSM write term. The output gate acts after temporal propagation and controls the residual path that mixes current features back into the temporal output. Mamba2's selective scan [4] uses data-dependent parameters; conditioned on a given input sequence, each step can still be viewed as following the write-and-propagate structure in Eq. 1. This conditioned SSM view focuses the analysis on the write path affected by the external mask. Fig. 3 illustrates the module architecture.

Input gating. We employ Mask-Aware Input Modulation:

$$x'_t = x_t \odot (1 - m_t), \tag{2}$$

where $m_t \in [0, 1]$ denotes the token-wise occlusion probability, broadcast over channels, and $\odot$ represents element-wise multiplication. The gate suppresses the input write term in Eq. 1:

$$\bar{B}_t x_t \;\rightarrow\; (1 - m_t)\, \bar{B}_t x_t. \tag{3}$$

Highly occluded tokens therefore contribute little to the state write, while visible tokens are written normally.

This placement follows the contamination recursion. Let $x_t^\star$ denote the clean observation, $x_t = x_t^\star + \eta_t$ the occlusion-perturbed observation, and ε_t the noise-induced state error. To isolate the explicit write path, we use a conditioned, linearized view in which the discretized parameters $\bar{A}_t$ and $\bar{B}_t$ are fixed around a trajectory. Without gating, the error evolves as

$$\varepsilon_t = \bar{A}_t \varepsilon_{t-1} + \bar{B}_t \eta_t. \tag{4}$$

With input gating, let D_t denote the diagonal or channel-broadcast operator induced by $(1 - m_t)$. Comparing the perturbed gated trajectory with its clean gated counterpart under the same local dynamics gives

$$\varepsilon_t^{\text{gate}} = \bar{A}_t \varepsilon_{t-1}^{\text{gate}} + D_t \bar{B}_t \eta_t. \tag{5}$$

When occlusion is severe ($m_t \to 1$), the entries of D_t approach zero and the write-path noise term is suppressed before it enters the state.

For a contiguous occlusion segment $[s, s{+}L{-}1]$ and a later frame $t \geq s{+}L{-}1$, define the transition product

$$\Phi(t,k) = \bar{A}_t \bar{A}_{t-1} \cdots \bar{A}_{k+1}, \qquad \Phi(t,t) = I. \tag{6}$$

Unrolling the gated error recursion over the segment yields

$$\varepsilon_t^{\text{gate}} = \Phi(t, s{-}1)\varepsilon_{s-1}^{\text{gate}} + \sum_{k=s}^{s+L-1} \Phi(t,k) D_k \bar{B}_k \eta_k. \tag{7}$$

Assume the conditioned dynamics are uniformly contractive, and the write matrices are bounded, with $\|\bar{A}_t\| \leq \alpha < 1$ and $\|\bar{B}_t\| \leq \beta$. If $\|\eta_k\| \leq \sigma$ on the segment and $\|D_k\| \leq \gamma_k \leq 1$, Eq. 7 gives

$$\|\varepsilon_t^{\text{gate}}\| \leq \alpha^{t-s+1} \|\varepsilon_{s-1}^{\text{gate}}\| + \beta\sigma \sum_{k=s}^{s+L-1} \alpha^{t-k} \gamma_k. \tag{8}$$

With $\gamma = \max_{k \in [s, s+L-1]} \gamma_k$, this becomes

$$\|\varepsilon_t^{\text{gate}}\| \leq \alpha^{t-s+1} \|\varepsilon_{s-1}^{\text{gate}}\| + \beta\sigma\gamma\, \alpha^{t-s-L+1} \frac{1-\alpha^L}{1-\alpha}. \tag{9}$$

The ungated case is recovered by setting $D_k = I$ and $\gamma = 1$. Under the conditioned, contractive write-path view above, a longer occlusion segment contributes a larger truncated geometric accumulation term, while input gating scales this term by the mask-dependent factor γ. The complete module, including the output gate, is evaluated empirically through ablation and controlled-contamination studies.

Output gating. Input gating protects the state write, but the residual connection can still reintroduce noisy input features after temporal propagation. We control the residual fusion ratio at the output stage using the occlusion probability M (a token-aligned obstacle mask, broadcast along channels). We summarize

occlusion severity as a scalar $\bar{m} = \mathrm{GAP}(M)$ and compute the residual retention ratio

$$\bar{m} = \mathrm{GAP}(M), \quad r = \sigma(\mathrm{MLP}(\bar{m})) \in (0, 1), \tag{10}$$

where r is then broadcast to match the shape of X_{in}. Let X_{in} denote the temporal block input and X_{out} the Mamba output:

$$X_{\text{base}} = X_{\text{in}} + X_{\text{out}}, \qquad X_{\text{hist}} = r \odot X_{\text{in}} + X_{\text{out}}, \tag{11}$$

$$X_{\text{time}} = (1 - M) \odot X_{\text{base}} + M \odot X_{\text{hist}}. \tag{12}$$

The gate is initialized to a small value, close to 0.2, at the beginning of training for stability. Under occlusion, it down-weights the contaminated input X_{in} while keeping the propagated output X_{out}. When $M = 1$ (full occlusion), the output reduces to $r \odot X_{\text{in}} + X_{\text{out}}$; when $M = 0$ (no occlusion), the standard residual $X_{\text{in}} + X_{\text{out}}$ is used.

Relation to Mamba internal selectivity. Mamba uses input-dependent parameters (Δ, B, C) learned from data for content-aware selectivity. Our dual-gating adds an obstacle-mask prior at spatial positions likely to contain corrupted observations. The two mechanisms operate at different levels: Mamba's internal selectivity acts on content representations, while the external gates use the obstacle mask to modulate state writing and residual fusion at occluded positions.

Multi-scale implementation. To aggregate information across the past T frames, we flatten the feature sequence $\{x_{t-T+1}, \ldots, x_t\}$ and occlusion sequence $\{m_{t-T+1}, \ldots, m_t\}$ along spatial positions to obtain $X \in \mathbb{R}^{(BHW)\times T\times C}$ and $M \in \mathbb{R}^{(BHW)\times T\times 1}$. A parallel low-resolution branch captures broader context: features are downsampled by a factor s, processed through Mamba with dual-gating, and upsampled back. The outputs are fused via a learnable residual:

$$\tilde{x}_t = x_{\text{out}} + \alpha_{\text{low}} \cdot \mathrm{Upsample}(\mathrm{Mamba}(\mathrm{Pool}_s(X))), \tag{13}$$

where α_{low} is initialized to a small value for stable training. The scale-pyramid branch combines fine local details with coarse context.

3.3 Lane-Guided Affine Warp

SSM-Consistent Dual-Gating applies temporal modeling after flattening spatial positions into token sequences. Each sequence therefore assumes that the same spatial index across frames refers to a consistent road location. Ego-motion and camera motion violate this assumption: a lane point or obstacle boundary may move to a different feature coordinate between adjacent frames. If such misaligned features are scanned along the T dimension, the SSM can mix evidence from different physical locations. The obstacle gate can also shift away from the feature region it is meant to modulate. We therefore align the historical feature, lane prior, and obstacle mask before temporal propagation, using the synchronous strategy shown in Fig. 4(a).

We use a local short-range affine transformation for lightweight 2D alignment between adjacent frames. This design matches the streaming state update, where information is propagated recursively and only the most recent transition must be synchronized before it enters the next temporal scan. In forward-facing driving videos, this low-dimensional parameterization is a compact approximation for synchronizing the feature, lane-prior, and obstacle-mask coordinates used by the gates. It is also less sensitive than dense optical flow [12] to textureless or occluded regions. The resulting alignment supplies the token-aligned obstacle prior used by dual-gating. The lane mask ℓ_{t-1} provides an additional geometric prior: masked global average pooling (GAP) focuses transformation estimation on structurally stable lane areas while ignoring cluttered backgrounds.

Let x_{t-1} and x_t denote adjacent frame features. The affine parameters are predicted as:

$$p = \tanh(\mathrm{MLP}([\mathrm{GAP}(x_{t-1}; \ell_{t-1}),\ \mathrm{GAP}(x_t)])), \tag{14}$$

where $[\cdot]$ denotes concatenation. We zero-initialize the final MLP layer so that the transformation starts from identity and avoids unstable early estimates. Reshaping p yields a residual affine matrix $\Delta\Theta \in \mathbb{R}^{B\times 2\times 3}$. The final affine matrix is obtained by adding it to the identity affine $I_{2\times 3}$:

$$\Delta\Theta = \mathrm{reshape}(p) \in \mathbb{R}^{B\times 2\times 3}, \qquad \Theta = I_{2\times 3} + \Delta\Theta. \tag{15}$$

The transformation synchronously warps features, lane masks, and obstacle masks via bilinear sampling:

$$\tilde{x}_{t-1} = \mathrm{warp}(x_{t-1}, \Theta), \quad \tilde{\ell}_{t-1} = \mathrm{warp}(\ell_{t-1}, \Theta), \quad \tilde{m}_{t-1} = \mathrm{warp}(m_{t-1}, \Theta). \tag{16}$$

This alignment keeps the occlusion signal $\tilde{m}_{t-1}$ and lane prior $\tilde{\ell}_{t-1}$ matched to the warped features $\tilde{x}_{t-1}$ and supports better aligned gating and structural guidance.

3.4 Structural Spatial Retrieval

Dual-gating suppresses contaminated writes and residuals (Sect. 3.2), but its temporal propagation can be insufficient when the historical state is weak and the current frame is heavily occluded. In this case, occluded regions still lack local lane evidence after gated temporal filtering. With aligned features $\tilde{x}_{t-1}$ and lane prior $\tilde{\ell}_{t-1}$ from Lane-Guided Affine Warp, SSR adds a spatial retrieval step to strengthen these missing structural cues. It uses cross-frame correspondence between the current frame and aligned history to recover structural evidence, as illustrated in Fig. 4(b).

Current-frame features serve as the query, while warped previous-frame features form the key and value. To guide attention toward lane-relevant regions, we inject a learnable lane-mask embedding into the key/value features:

$$Q = \mathrm{LN}(x_t), \qquad K = V = \mathrm{LN}(\tilde{x}_{t-1}) + \mathrm{LN}(e(\tilde{\ell}_{t-1})), \tag{17}$$

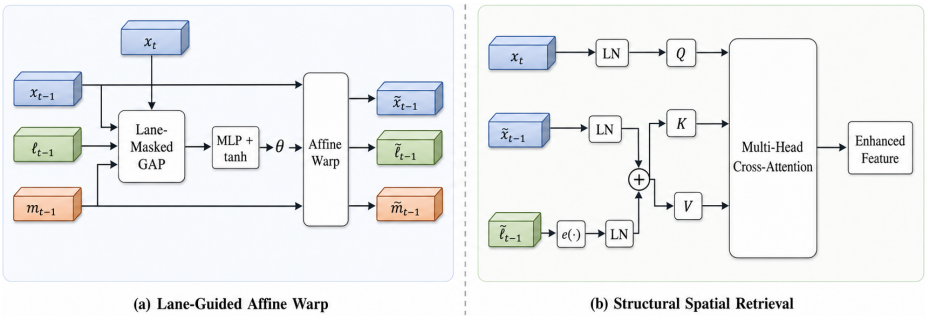

Fig. 4: Lane-Guided Affine Warp and Structural Spatial Retrieval. (a) In Lane-Guided Affine Warp, features x_{t-1} and lane mask ℓ_{t-1} are fed into Lane-Masked GAP to compute lane-region-weighted statistics. Combined with current-frame features x_t, an MLP predicts affine parameters p, where tanh constrains the transformation magnitude. The Affine Warp operator synchronously transforms x_{t-1}, ℓ_{t-1}, and m_{t-1} using the same grid, yielding aligned outputs $\tilde{x}_{t-1}$, $\tilde{\ell}_{t-1}$, and $\tilde{m}_{t-1}$. (b) In Structural Spatial Retrieval, current-frame features x_t pass through LN to form the query Q. Warped features $\tilde{x}_{t-1}$ and lane-mask embedding $e(\tilde{\ell}_{t-1})$ are normalized via LN and summed to form key K and value V. Multi-head cross-attention produces the enhanced feature output.

where LN denotes Layer Normalization and $e(\cdot)$ is a lightweight convolutional network mapping the single-channel mask $\tilde{\ell}_{t-1}$ to a C-dimensional feature space. Cross-attention follows the standard multi-head formulation [25]:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) V. \tag{18}$$

The $QK^\top$ similarity matrix computes soft spatial correspondence, so each query position can retrieve structural cues from history without explicit offset prediction. Injecting the lane-mask embedding into the key/value features biases attention toward lane regions. Positions with stronger lane priors can receive higher matching scores, improving cross-frame completion under occlusion. SSR works with Dual-Gating (Sect. 3.2), which handles recursive temporal state propagation.

3.5 Geometry-Aware State Initialization

The previous modules control how observations enter an already running temporal state. At the beginning of a stream, however, this state is still uninitialized. Zero initialization ($h_0 = 0$) makes the first few state updates depend strongly on early frame observations, so an occluded or ambiguous first frame can dominate the initial memory. We use Geometry-Aware State Initialization to give the SSM a weak spatial prior before any image observation is written into the state. We prepend a learnable start token s to each spatial token sequence before Mamba processing. This token is read before the video frames and is not attenuated by

Fig. 5: Visualization of the spatial prior learned by Geometry-Aware State Initialization. Each strip shows the input image, position-prior heatmap, and ground-truth lanes. The learned prior is concentrated around lane-like structures and provides a cold-start bias before reliable temporal history is available.

the obstacle mask. To give it spatial semantics, we parameterize it as the sum of a context-agnostic shared token and a position-specific projection:

$$s = \text{token}_{\text{shared}} + \text{Proj}(\text{PE}_{2D}), \tag{19}$$

where $s \in \mathbb{R}^{(BHW)\times 1\times C}$ and PE_{2D} encodes normalized 2D coordinates. By processing $X^+ = \text{concat}(s, X)$, the SSM updates its hidden state $h_{-1} \to h_0$ using s before seeing any video frames. This initialization complements dual-gating: the gates suppress contaminated writes, while the start token provides a geometry-aware state from which reliable temporal propagation can begin. Fig. 5 visualizes this learned prior, which tends to concentrate around lane-like regions.

4 Experiments

4.1 Implementation Details

The base detector described in Sect. 3.1 is initialized from the pretrained OMR checkpoint and kept frozen. Only the temporal aggregation modules are trained. Following OMR, obstacle masks are produced by a frozen SegFormer-B5 [30]; the segmenter is not updated during training. The temporal module uses Mamba2 [4] with an 8-frame memory window. It is trained with the same lane classification and coefficient regression losses as the base detector. Detailed hyperparameters, dataset descriptions, and metric definitions are provided in the supplementary material.

4.2 Datasets and Evaluation Metrics

We evaluate SIGMA-Lane on VIL-100 [35] and OpenLane-V [12], two video lane benchmarks with persistent vehicle occlusion and diverse driving conditions. Following the CULane protocol [19], we report per-frame quality using mIoU, accuracy, and F1 at IoU thresholds 0.5 and 0.8. We also report temporal flickering and missing rates R_F/R_M from [12], where lower values indicate fewer inconsistent detections across adjacent frames. The supplementary material gives the dataset details and metric definitions.

Table 1: Comparison on VIL-100. Best results are in **bold**, second best are underlined.

Method	Approach	mIoU(↑)	F1@0.5(↑)	F1@0.8(↑)	Accuracy(↑)	R_F@0.5(↓)	R_M@0.5(↓)
LaneNet [18]	Image-based	0.633	0.721	0.222	0.858	–	–
LSTR [16]		0.573	0.703	0.131	0.884	–	–
LaneATT [23]		0.664	0.823	–	0.912	–	–
DiLane [3]		0.745	0.837	0.505	0.884	–	–
MMA-Net [35]	Video-based	0.705	0.839	0.458	0.910	0.042	0.127
MLM-Net [28]		0.753	0.904	**0.624**	–	–	–
RVLD [12]		0.787	0.924	0.582	–	0.038	0.050
PHNet [2]		0.783	0.915	0.615	0.908	–	–
OMR [11]		0.774	0.936	0.504	0.948	0.026	0.038
LaneTCA [38]		0.796	0.933	0.621	0.951	0.039	0.055
SIGMA-Lane	Video-based	**0.801**	**0.940**	0.595	**0.956**	**0.023**	**0.030**

4.3 Comparative Results

VIL-100. Tab. 1 compares SIGMA-Lane with image-based [3, 16, 18, 23] and video-based [2,11,12,28,35,38] detectors. SIGMA-Lane obtains the best F1@0.5 (0.940), accuracy (0.956), and mIoU (0.801). Although SIGMA-Lane does not obtain the best F1@0.8, it has the lowest flickering and missing rates on VIL-100. Compared with LaneTCA, it reduces R_F/R_M by 41.0%/45.5%. It also improves F1@0.8 by 0.091 over OMR, suggesting greater robustness when heavy occlusions suppress per-frame evidence. Fig. 6 shows examples where SIGMA-Lane preserves lane continuity when large vehicles obscure multiple lanes.

OpenLane-V. Tab. 2 reports results on the larger OpenLane-V benchmark. LaneTCA [38] obtains the highest mIoU (0.774), while SIGMA-Lane obtains the highest F1@0.5/F1@0.8 (0.838/0.591) and the lowest persistent missing rate

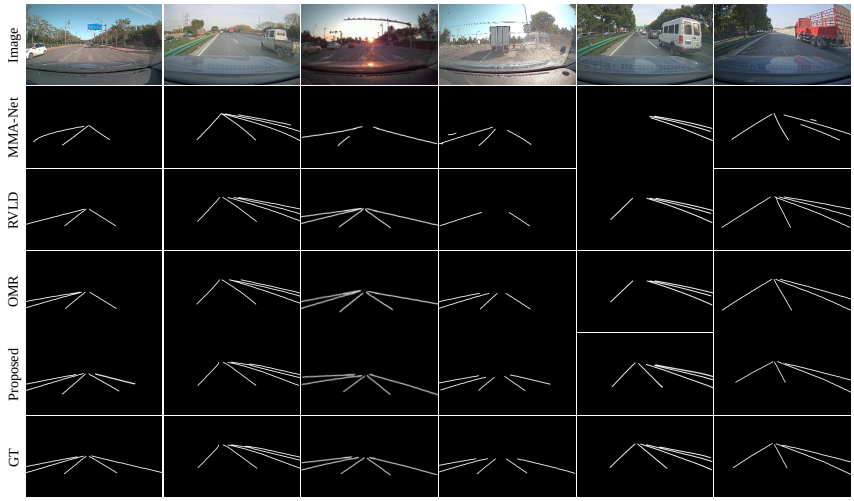

Fig. 6: Comparison of lane detection results on the VIL-100 dataset.

Table 2: Comparison on OpenLane-V.

Method	Approach	mIoU(↑)	F1@0.5(↑)	F1@0.8(↑)	R_F@0.5(↓)	R_M@0.5(↓)	R_F@0.8(↓)	R_M@0.8(↓)
MFIALane [21]		0.697	0.723	0.475	0.061	0.300	0.080	0.519
CondLaneNet [15]		0.698	0.780	0.450	0.047	0.239	0.084	0.531
GANet [26]	Image-based	0.716	0.801	0.530	0.048	0.198	0.082	0.443
CLRNet [37]		0.735	0.789	0.554	0.054	0.224	0.086	0.430
DiLane [3]		0.740	0.795	0.573	0.043	0.232	0.082	0.409
ConvLSTM [41]		0.529	0.641	0.353	0.058	0.282	0.091	0.574
ConvGRUs [34]		0.540	0.641	0.355	0.064	0.288	0.094	0.576
MMA-Net [35]		0.574	0.573	0.328	0.044	0.461	0.071	0.671
RVLD [12]	Video-based	0.727	0.825	0.566	**0.014**	0.167	**0.051**	0.406
OMR [11]		0.742	0.836	0.582	0.016	**0.162**	0.055	0.393
PHNet [2]		0.757	0.804	0.588	0.025	0.196	0.064	0.399
LaneTCA [38]		**0.774**	0.822	0.574	0.050	0.212	0.084	0.424
SIGMA-Lane	Video-based	0.761	**0.838**	**0.591**	0.018	**0.162**	0.052	**0.384**

at the stricter threshold (R_M@0.8 = 0.384). This pattern reflects the different operating modes of the methods: LaneTCA aggregates a temporal window and achieves stronger average overlap, whereas SIGMA-Lane performs streaming state propagation and explicitly filters occluded writes. The gains at the stricter threshold match the intended use case: preserving valid lane instances under occlusion. Fig. 7 visualizes results across diverse conditions, and Fig. 8 traces one occluded case from intermediate maps to the recovered lane prediction. Additional examples of frame-to-frame behavior are provided in the supplementary material.

Fig. 7: Comparison of lane detection results on the OpenLane-V dataset.

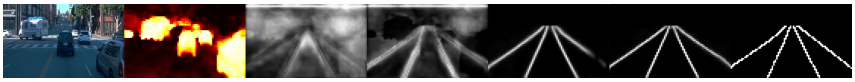

Fig. 8: Qualitative analysis of SIGMA-Lane under vehicle occlusion. Current-frame maps (F_t, P_t) show weakened lane responses in the occluded region, while refined maps $(\tilde{F}_t, \tilde{P}_t)$ recover the lane topology and produce a prediction closer to GT. Maps are min-max normalized.

4.4 Computational Cost

Because SIGMA-Lane targets streaming video inference, we compare the computational cost of representative video lane detectors under the same 384×640 input setting in Tab. 3. All timing is measured on a single NVIDIA RTX 4090 in full-validation mode, using 50 warmup frames and all remaining frames for measurement. Model FPS measures the network-only forward pass, while E2E FPS includes post-processing and NMS inside the per-frame inference loop. Main Params exclude auxiliary ILD models and count only the temporal model used in the final stage. Agg. Params count only temporal aggregation modules. SIGMA-Lane has the lowest FLOPs and temporal aggregation parameters in this comparison, and runs faster than LaneTCA. OMR gives the highest throughput; SIGMA-Lane uses fewer FLOPs and aggregation parameters while obtaining higher accuracy.

Table 3: Computational cost comparison on OpenLane-V. Model FPS denotes network-only forward speed, while E2E FPS includes post-processing and NMS.

Method	FLOPs(G)(↓)	Model FPS(↑)	E2E FPS(↑)	Main Params(M)	Agg. Params(M)
OMR [11]	54.6	**215.96**	**141.15**	7.15	3.54
LaneTCA [38]	28.9	41.57	28.48	**1.62**	0.96
SIGMA-Lane	**27.1**	118.79	96.74	3.96	**0.35**

4.5 Ablation Studies

We next isolate each component in the contamination-aware aggregation module. Row 1 in Tab. 4 replaces OMR's ConvLSTM with a standard Mamba2 [4] operator, with no gating or alignment. This gives a direct backbone-swap reference.

Dual-gating (rows 1–4). The baseline (row 1) is a competitive temporal backbone-swap reference. Input gating (row 2) attenuates the write term $(1-m_t)\bar{B}_t x_t$ to reduce corrupted state writes. Output gating (row 3) reduces the residual contribution of the noisy current frame at the fusion stage. Combining both gates (row 4) gives +0.012 mIoU (0.775→0.787), which accounts for 46% of the total mIoU gain (0.775→0.801) in Tab. 4. Dual-gating also reduces

Table 4: Ablation studies on VIL-100. DG-I/O: Input/Output Gating, Warp: Affine Warp, SSR: Structural Spatial Retrieval, Token: Geometry-Aware Token.

#	DG-I	DG-O	Warp	SSR	Token	mIoU(↑)	F1@0.5(↑)	Accuracy(↑)	R_F@0.5(↓)	R_M@0.5(↓)
1						0.775	0.926	0.942	0.052	0.084
2	✓					0.781	0.930	0.947	0.036	0.045
3		✓				0.778	0.929	0.946	0.040	0.053
4	✓	✓				0.787	0.935	0.950	0.024	0.032
5	✓	✓	✓			0.794	0.937	0.952	0.025	0.035
6	✓	✓	✓	✓		0.798	0.938	0.953	0.024	0.033
7	✓	✓	✓	✓	✓	**0.801**	**0.940**	**0.956**	**0.023**	**0.030**

R_F/R_M from 0.052/0.084 to 0.024/0.032. The full model reaches 0.023/0.030 after adding alignment, retrieval, and initialization.

Lane-guided affine warp (row 5). Adding synchronous alignment gives +0.007 mIoU and keeps the obstacle mask m_t aligned with the regions it should modulate. Without it, misalignment between warped features and the static mask can leak contaminated signals or over-suppress clean areas, which also hurts F1 and accuracy.

Structural Spatial Retrieval (row 6). SSR adds +0.004 mIoU by retrieving lane-relevant spatial cues from warped historical features via lane-mask-guided cross-attention. It supplies structural information that gating alone cannot reconstruct, and also improves F1@0.5 and accuracy.

Geometry-aware initialization (row 7). With dual-gating, alignment, and retrieval already enabled, the start token further improves the reliability of the initial temporal state. Row 7 gives the best values across all reported metrics, including the lowest missing rate, indicating that a geometry-aware prior helps recurrent aggregation start from lane-relevant spatial structure before sufficient history has accumulated.

Ablation summary. These component ablations indicate a clear division of labor. Dual-gating provides the main temporal-stability gain by suppressing contaminated writes and residual fusion, while alignment, SSR, and the start token improve spatial recovery and initialization.

4.6 Contamination and Occlusion Analysis

We further analyze whether the improvement comes from the proposed contamination-aware use of obstacle priors, especially under corrupted memory and heavier occlusion.

Mask and gate sensitivity. Tab. 5 separates two possible explanations: whether the gain comes from having an obstacle prior, or from how the temporal model uses that prior. Perturbing the mask has a limited effect. A 7×7 dilation changes F1/mIoU by only $-0.0014/-0.0010$, and erasing 20% of obstacle pixels changes them by $+0.0007/-0.0005$. By contrast, removing both gates while keeping the original mask drops F1/mIoU by $-0.0180/-0.0150$ and increases R_M by +0.0740. The result indicates that mask precision is not the main

Table 5: Mask and gate sensitivity on OpenLane-V validation. Deltas are measured relative to clean SIGMA-Lane. Lower is better for ΔR_F and ΔR_M.

Setting	ΔF1(↑)	ΔmIoU(↑)	ΔR_F(↓)	ΔR_M(↓)
No DG-I/O; original M	−0.0180	−0.0150	+0.0362	+0.0740
Full SIGMA; dilate M (7×7)	−0.0014	−0.0010	+0.0012	+0.0024
Full SIGMA; erase 20% M=1	+0.0007	−0.0005	+0.0008	−0.0010

factor; the critical operation is placing the prior on the write and residual-fusion paths.

Controlled contamination and occlusion severity. The supplementary material reports two additional analyses under stronger stress. Structured memory perturbations along occlusion boundaries increase R_M@0.5 from 0.162 to 0.182 for SIGMA-Lane, but from 0.236 to 0.367 for the ungated control, a 6.6× larger degradation without the gates. In the occlusion-stratified evaluation on OpenLane-V, SIGMA-Lane and OMR are comparable in Light frames, while SIGMA-Lane gains +0.028 mIoU over OMR in both Moderate and Heavy bins.

5 Conclusion

This paper studies video lane detection under persistent occlusion from the perspective of SSM state updates. We identify *state contamination*: occlusion-corrupted observations can be written into temporal memory and affect later predictions. SIGMA-Lane addresses this failure mode by gating the SSM write and residual-fusion paths, with coordinate-consistent alignment, SSR, and geometry-aware initialization supporting temporal propagation and structural recovery. Experiments on VIL-100 and OpenLane-V show improved temporal stability with competitive per-frame detection quality. The ablations, mask sensitivity analysis, and controlled contamination tests indicate that the obstacle prior is most useful when it directly controls how information enters and leaves the temporal update. The current design operates in 2D image-feature space, where affine warping provides a local short-range alignment approximation. A natural extension is to combine contamination-aware state updates with 3D geometric cues, camera pose, or depth-aware motion for more complex driving scenes.

Acknowledgements This work was supported by the National Natural Science Foundation of China (Grant Nos. 62403429, 62476247, and 62402442), the Hangzhou Key Research and Development Program (Grant No. 2025SZDA0100), the Zhejiang Provincial Natural Science Foundation of China (Grant Nos. LQN25F030008 and LQ24F020038), and the Fundamental Research Funds for the Provincial Universities of Zhejiang (Grant No. G26081190006).

References

1. Chen, M., Jia, Q., Yang, J., Liu, S.: SR-LMamba: A lane detection model for complex scenes integrating curvelet transform with Mamba architecture. PLOS ONE **20** (2025)
2. Cheng, Z., Wang, C., Zhang, G., Zhou, W.: Parallel heterogeneous networks with adaptive routing for online video lane detection. IEEE Trans. Intell. Transp. Syst. **26**(4), 5225–5235 (2025)
3. Cheng, Z., Zhang, G., Wang, C., Zhou, W.: DILane: Dynamic instance-aware network for lane detection. In: ACCV. pp. 2075–2091 (2022)
4. Dao, T., Gu, A.: Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In: ICML. pp. 10041–10071 (2024)
5. Feng, Z., Guo, S., Tan, X., Xu, K., Wang, M., Ma, L.: Rethinking efficient lane detection via curve modeling. In: CVPR (2022)
6. Gu, A., Dao, T.: Mamba: Linear-time sequence modeling with selective state spaces. arXiv preprint arXiv:2312.00752 (2023)
7. Gu, A., Goel, K., Ré, C.: Efficiently modeling long sequences with structured state spaces. In: ICLR (2022)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
9. Hou, Y., Ma, Z., Liu, C., Hui, T.W., Loy, C.C.: Inter-region affinity distillation for road marking segmentation. In: CVPR (2020)
10. Hou, Y., Ma, Z., Liu, C., Loy, C.C.: Learning lightweight lane detection CNNs by self attention distillation. In: ICCV (2019)
11. Jin, D., Kim, C.S.: OMR: Occlusion-aware memory-based refinement for video lane detection. In: ECCV. pp. 129–145 (2024)
12. Jin, D., Kim, D., Kim, C.S.: Recursive video lane detection. In: ICCV. pp. 8473–8482 (2023)
13. Jin, D., Park, W., Jeong, S.G., Kwon, H., Kim, C.S.: Eigenlanes: Data-driven lane descriptors for structurally diverse lanes. In: CVPR (2022)
14. Li, K., Li, X., Wang, Y., He, Y., Wang, Y., Wang, L., Qiao, Y.: VideoMamba: State space model for efficient video understanding. arXiv preprint arXiv:2403.06977 (2024), eCCV 2024
15. Liu, L., Chen, X., Zhu, S., Tan, P.: CondLaneNet: A top-to-down lane detection framework based on conditional convolution. In: ICCV (2021)
16. Liu, R., Yuan, Z., Liu, T., Xiong, Z.: End-to-end lane shape prediction with transformers. In: WACV (2021)
17. Liu, Y., Tian, Y., Zhao, Y., Yu, H., Xie, L., Wang, Y., Ye, Q., Jiao, J., Liu, Y.: VMamba: Visual state space model. arXiv preprint arXiv:2401.10166 (2024), neurIPS 2024
18. Neven, D., De Brabandere, B., Georgoulis, S., Proesmans, M., Van Gool, L.: Towards end-to-end lane detection: An instance segmentation approach. In: Intelligent Vehicles Symposium (2018)
19. Pan, X., Zhan, X., Shi, J., Luo, P., Wang, X., Tang, X.: Spatial as deep: Spatial CNN for traffic scene understanding. In: AAAI. pp. 7276–7283 (2018)
20. Qin, Z., Wang, H., Li, X.: Ultra fast structure-aware deep lane detection. In: ECCV (2020)
21. Qiu, Z., Zhao, J., Sun, S.: Mfialane: Multiscale feature information aggregator network for lane detection. IEEE Transactions on Intelligent Transportation Systems **23**(12), 24263–24275 (2022)

22. Su, H., Wu, W., Song, F., Zhang, J., Yang, Z., Yan, J.: DriveMamba: Task-centric scalable state space model for efficient end-to-end autonomous driving. In: ICLR (2026)
23. Tabelini, L., Berriel, R., Paixao, T.M., Badue, C., De Souza, A.F., Oliveira-Santos, T.: Keep your eyes on the lane: Real-time attention-guided lane detection. In: CVPR (2021)
24. Tabelini, L., Berriel, R., Paixao, T.M., Badue, C., De Souza, A.F., Oliveira-Santos, T.: PolyLaneNet: Lane estimation via deep polynomial regression. In: ICPR. pp. 6150–6156 (2021)
25. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017)
26. Wang, J., Ma, Y., Huang, S., Hui, T., Wang, F., Qian, C., Zhang, T.: A keypoint-based global association network for lane detection. In: CVPR (2022)
27. Wang, M., Zhang, Y., Feng, W., Zhu, L., Wang, S.: Video instance lane detection via deep temporal and geometry consistency constraints. In: ACM MM (2022)
28. Wang, X., Yin, Y., Huang, F., Bao, X.: MLM-Net: Streamlined multi-lane detection network with spatio-temporal memory for video instance lane detection. In: IEEE Int. Conf. Intell. Transp. Syst. (2023)
29. Xiao, L., Li, X., Yang, S., Yang, W.: ADNet: Lane shape prediction via anchor decomposition. In: ICCV (2023)
30. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: SegFormer: Simple and efficient design for semantic segmentation with transformers. In: NeurIPS. pp. 12077–12090 (2021)
31. Xu, H., Wang, S., Cai, X., Zhang, W., Liang, X., Li, Z.: CurveLane-NAS: Unifying lane-sensitive architecture search and adaptive point blending. In: ECCV (2020)
32. Xu, S., Cai, X., Zhao, B., Zhang, L., Xu, H., Fu, Y., Xue, X.: RCLane: Relay chain prediction for lane detection. In: ECCV (2022)
33. You, Z., Wang, N., Wang, H., Zhao, Q., Wang, J.: MambaBEV: An efficient 3D detection model with Mamba2. arXiv preprint arXiv:2410.12673 (2025)
34. Zhang, J., Deng, T., Yan, F., Liu, W.: Lane detection model based on spatio-temporal network with double convolutional gated recurrent units. IEEE Trans. Intell. Transp. Syst. **23**(7), 6666–6678 (2021)
35. Zhang, Y., Zhu, L., Feng, W., Fu, H., Wang, M., Li, Q., Li, C., Wang, S.: VIL-100: A new dataset and a baseline model for video instance lane detection. In: ICCV. pp. 15681–15690 (2021)
36. Zheng, T., Fang, H., Zhang, Y., Tang, W., Yang, Z., Liu, H., Cai, D.: RESA: Recurrent feature-shift aggregator for lane detection. In: AAAI (2021)
37. Zheng, T., Huang, Y., Liu, Y., Tang, W., Yang, Z., Cai, D., He, X.: CLRNet: Cross layer refinement network for lane detection. In: CVPR. pp. 898–907 (2022)
38. Zhou, K., Li, L., Zhou, W., Wang, Y., Feng, H., Li, H.: LaneTCA: Enhancing video lane detection with temporal context aggregation. IEEE TCSVT **35**(9), 8574–8585 (2025)
39. Zhou, K.: Lane2seq: Towards unified lane detection via sequence generation. In: CVPR. pp. 16944–16953 (2024)
40. Zhu, L., Liao, B., Zhang, Q., Wang, X., Liu, W., Wang, X.: Vision mamba: Efficient visual representation learning with bidirectional state space model. In: ICML (2024)
41. Zou, Q., Jiang, H., Dai, Q., Yue, Y., Chen, L., Wang, Q.: Robust lane detection from continuous driving scenes using deep neural networks. IEEE Trans. Veh. Technol. **69**(1), 41–54 (2020)