Journal of Applied Automation Technologies

Research Paper

Interaction-Behaviour-Driven Lancin-Mamba for Trajectory Prediction at Unsignalized Intersections

Oliver Jõgi[1, *], Taavi Rannik[1] and Egert Võsa[1]

[1] Department of Computer Networks, Rakvere Vocational College, Rakvere, 44310, Estonia
*Corresponding author: oliver.j@rvk.ee

Abstract. Rather than merely extending individual kinematics, accurate motion forecasting at unsignalized junctions must take into account the reasons behind driver behavior, such as adjusting to priority ambiguity, perceived gaps, and anticipated actions by other drivers. An interaction-behavior-driven Lancin-Mamba model with lane-topology reasoning and selective state-space sequence modeling is presented in this study. First, create a conflict-aware interaction graph based on lane conflict sites, relative arrival times, approach priority, and observed yielding data. Long observation histories are summarized by a bidirectional selective Mamba block without the quadratic temporal penalty of self-attention, and Lancin presents agent-to-lane and lane-to-lane constraints. The multimodal decoder is reduced by the fused representation to produce mode probabilities and route-consistent trajectories. A differentiable conflict loss lowers simultaneous occupancy close to shared conflict spots, while a behavior-consistency goal penalizes futures that deviate from the anticipated yield/pass connection. The suggested model lowers the 6-s minimum final displacement error from 2.31 m to 1.92 m and the miss rate from 13.8% to 10.4% when compared to a strong Lancin baseline, according to experiments conducted on a controlled unsignalized-intersection benchmark with packed, moderate, and sparse traffic regimes. maintained scene delay at 31.7 ms while reducing the anticipated conflict frequency by 28.6%. The explicit interaction edges are responsible for the greatest increase in accuracy, whereas selective state-space temporal modeling is responsible for the greatest efficiency gain. Based on the aforementioned tests, a tiny predictor for planning-oriented intelligent vehicle systems can incorporate both structured map context and behavioral negotiation.

Keywords: *Unsignalized Intersection, Trajectory Prediction, Interaction Behavior, Lancin, Mamba, Selective State-space Model*

Received on 24 December 2023, Accepted on 19 September 2024, Published on 03 October 2024

Introduction

Drivers have fewer options at a single location at unsignalized intersections. Before the decision is visible as a finished move, a vehicle must determine a viable lane continuation, estimate which agents can reach a shared conflict site, read incomplete priority cues, and modify acceleration. Through the aforementioned procedures, occlusion and various driving styles can result in a firm pass, a rolling yield, or a complete halt in geometrically identical circumstances. Therefore, conditional multi-agent forecasting takes a relational approach to the problem, which means that an actor's future depends on what that actor anticipates from others [1]. Additionally, the planning-oriented predictor ought to take into account lane topology, retain store behavior data for a brief period of time, and display multiple feasible futures. Static proximity graphs lack the conflict-point, priority, and progress indicators that influence real-world negotiation, despite Lancin's superb graph-based approach of organized road design [2].

The geometric structure is enhanced by extended-context temporal modeling. Extended sequences are summarized in a compact recurring state by structured state-space models, which can highlight sparse intent evidence like brake onsets or hesitation zone entry [3]. Lane connections and joint compatibility cannot be stored in temporal memory on its own. Even moderately reasonable trajectories are mutually contradictory, as

https://doi.org/10.64972/jaat.2024v2.326p31e:427-442

demonstrated via interactive factorization [4]. In addition to endpoint error, safety compatibility evaluation is also motivated by planning-oriented representations [5]. Calibration and scene-level consistency evaluation are also motivated by large-scale joint-motion benchmarks [6]. Thus, the initial problem is how to combine multimodal decoding, efficient temporal memory, directed interaction behavior, and lane layout into a single predictor.

In this project, create an interaction-behavior-driven Lancin-Mamba architecture for trajectory prediction at unsignalized crossings. Bidirectional selected state-space blocks summarize each actor's observation history, a dynamic conflict-aware graph model yields/passes negotiation, and lane graph convolutions encode reachable centerlines and actor-map links. To reduce incompatible futures, behavior-consistency and differentiable conflict losses are used to train a route-consistent multimodal decoder. Based on traffic density, maneuver class, and arrival-time gap, the evaluation divides displacement accuracy, conflict incidence, calibration, robustness, and latency; ablations are used to ascertain each component's contribution.

Related Work

Map-constrained trajectory prediction

Rasterized bird's-eye view inputs have given way to vector and graph representations in map-conditioned forecasting. Although raster encoders have a resolution constraint and can blur narrow lane boundaries due to repeated convolutions, they are appropriate for heterogeneous situations. For route-consistent decoding, Vector Net [7] can add semantics to road elements and supports polylines. Information spreads along the road network rather than over the image because lane graphs can display predecessor, successor, and lateral links.

Because the attainable pathways abruptly divide at the intersection, Lancin-style reasoning is more appropriate. Lane-to-lane propagation broadens the receptive field, actor-to-lane fusion aligns motion with local segments, and lane-to-actor fusion provides dynamic agents with topological context. The aforementioned concept is expanded to actor-centric lane regions and distributed graph representations in Lantern [8]. Although radius-based neighborhoods are comparatively straightforward, they are unable to differentiate between distant proximity and real crossing issues.

Typically, multimodal decoders regress a trajectory for each mode after producing multiple end-points or latent motion modes. Cover Net frames the task as classification over a trajectory set [9], while Multipath++ employs learnt trajectory anchors [10]. Although map losses discourage off-road outputs and diversity aims lessen collapse, two independently possible paths may nonetheless be jointly inconsistent if they simultaneously inhabit the same conflict zone.

Interaction and behavior modeling

Social pooling, graph attention, relation networks, and latent-variable models have all been used in interaction-aware predictors. Pooling-based models may lose pair identity, while Goal-GAN [11] builds multimodality on interpretable goal locations. While pairwise selection is still carried out by attention, geometrically irrelevant associations might be given weights.

Behavioral variables include things like yield, pass, follow, and combine. Agent Former uses agent-aware attention to jointly represent time and social influence [12]. Although latent techniques eliminate manual annotation, their modes could not match operationally important correlations. Explicit behavior labels help stabilize training but are expensive and unclear in early observation.

The coupling is made clearer by Joint Scene Models. Way former suggests a unified attention architecture for scene-level motion prediction [13], while SMART uses recurrent state pooling to create scene-consistent multi-agent futures [14]. Although the presumption of independence is lessened, these designs still need learning the interaction structure from a large-scale context.

Losses that are safety-conscious avoid crashes or necessitate social compatibility. By identifying global intentions and carrying out local adjustments, Motion Transformer [15] improves multimodal decoding; nonetheless, conflict-specific timing remains a problem. Conflict-point occupancy concentrates the penalty at the actual path crossing, whereas hard distance limitations may unduly penalize legitimate near-passes.

https://doi.org/10.64972/jaat.2024v2.326p31e:427-442

Long-context sequence models

Over extended observations, recurrent networks lose early information despite having a tiny temporal state. Transformer attention makes direct connections between far-off events, while temporal convolution has a fixed kernel and expands the receptive field. The aforementioned techniques provide decent starting points, but as actors, map elements, and history length multiply, memory and latency will rise.

Structured State-Space Models use learnt dynamical state changes to summarize a series. When there is little conclusive data regarding steady motion, linear-sequence scaling is appropriate for high-rate on-board inference. The remaining gap is a model that links lane topology and explicit negotiation structure to long-memory updates rather than another general-purpose temporal backbone.

Problem Formulation and Interaction Encoding

Scene representation and prediction task

A scene contains N dynamic agents observed for T_o samples and a vector map composed of directed lane segments. For each agent, the input state includes planar position, velocity, acceleration, heading, and observation validity in an agent-centered coordinate system. Map segments carry centerline geometry, turn type, control status, and adjacency. The predictor outputs K future hypotheses over T_f samples with a probability for each hypothesis. Training scenes are centered on a target agent, but all actors whose reachable lane paths intersect the target corridor are retained. This avoids a fixed-radius cutoff that could omit a fast vehicle approaching from a distant arm.

$$X = \{\mathbf{x}_i^t\}_{i=1,t=-T_o+1}^{N,0} \quad \text{Eq.(1)}$$

Here X denotes the observed scene history. The state vector for agent i at time t contains position, velocity, acceleration, heading, lane-relative coordinates, and an observation-validity flag. N varies by scene, whereas the observation length T_o is fixed at 20 samples.

$$Y^i = \{p_i^k, \hat{\mathbf{y}}_{i,\tau}^k\}_{k=1}^K, \sum_{k=1}^K p_i^k = 1 \quad \text{Eq.(2)}$$

The predicted output for agent i contains K trajectory hypotheses and their normalized mode probabilities. Each hypothesis spans T_f future samples. The probability simplex supports likelihood and calibration evaluation, while the trajectory set supports best-of-K displacement metrics.

Coordinate normalization follows the target-centered forecasting convention used by Argo verse [16], placing the target at the origin and aligning its final observed heading with the local x direction. The same rigid transform is applied to every neighbor and map element. Sensor-rich datasets such as nosecones [17] also motivate an explicit validity mask so that missing observations are not confused with stopped motion. Figure 1 summarizes how observed actors, reachable lane segments, and shared conflict points become the structured input.

Conflict-aware behavior graph

Candidate interaction pairs are found by intersecting their reachable lane polylines. For every valid pair, the nearest shared conflict point is retained together with path distance from each agent to that point. Relative arrival time, normalized approach speed, priority cue, and signed longitudinal acceleration form an edge feature. The construction distinguishes crossing, merging, following, and non-conflicting relations before learning. It also keeps directed edges because evidence from a yielding actor and evidence from a proceeding actor have different meanings.

$$\Delta t_{ij} = \frac{d_i{}^c}{v_i + \varepsilon} - \frac{d_j{}^c}{v_j + \varepsilon} \quad \text{Eq.(3)}$$

The relative arrival-time feature compares estimated travel time from agents i and j to their shared conflict point c. Distances are measured along reachable lane paths, v is approach speed, and the small constant ε prevents numerical instability near zero speed. Its sign indicates nominal arrival order.

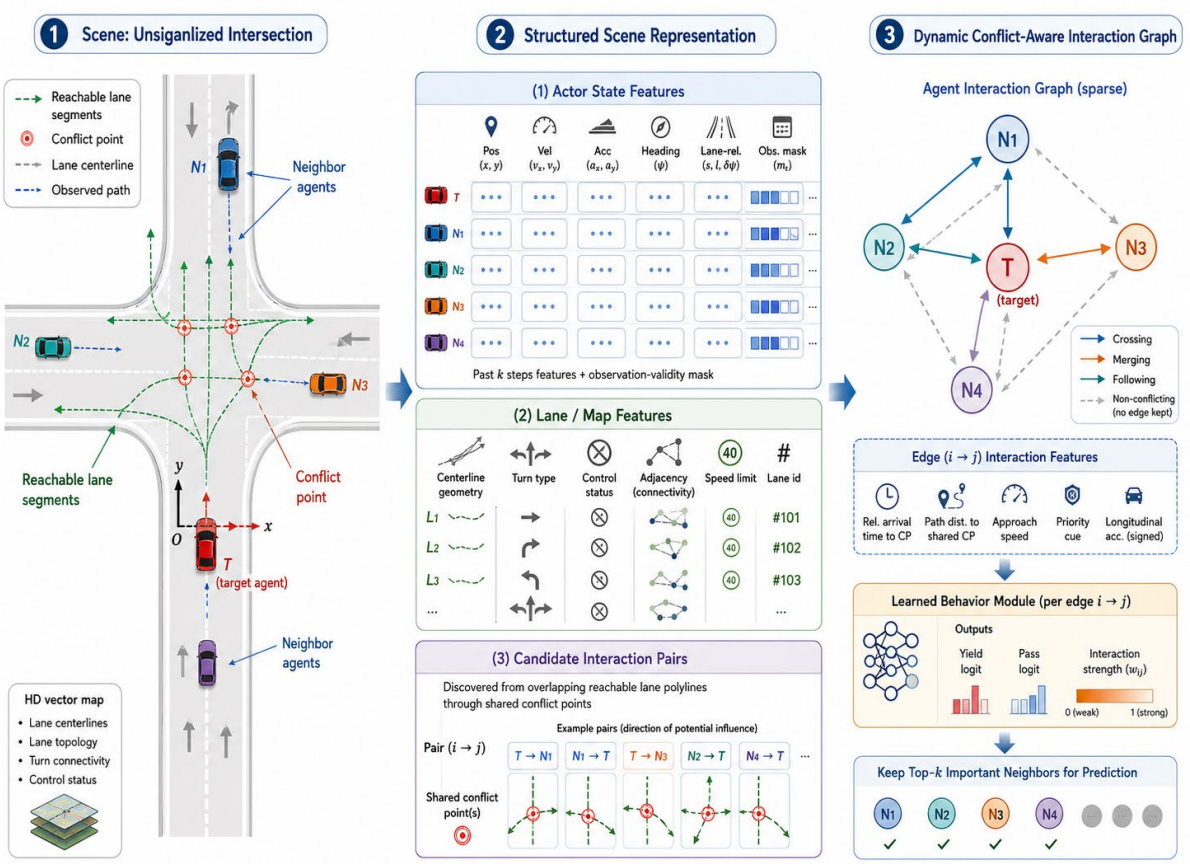

Figure 1. Conflict-point scene representation and dynamic interaction-graph construction.

The edge weight combines geometric exposure with behavioral evidence. A small predicted arrival-time gap raises interaction strength, while clear lane separation or diverging motion suppresses it. Sustained negative acceleration increases the probability that the source yields; positive progress through the hesitation zone increases the probability that it proceeds. A small multilayer network converts these signals into soft yield/pass logits. Soft labels are derived from actual conflict-point entry order during training but are not used at inference.

$$e_{ij}^{t} = \sigma(f_e([h_i^t, h_j^t, \Delta t_{ij}^t, q_{ij}, a_i^t, a_j^t])) \quad \text{Eq.(4)}$$

The directed interaction strength is produced by a learned edge function followed by a sigmoid. Its inputs include both actor states, the relative arrival time, a conflict-and-priority descriptor, and signed accelerations. The resulting scalar gates messages from agent j to agent i at every observation step.

Graph sparsity is controlled by retaining at most six high-exposure neighbors per actor. This cap covers simultaneous conflicts on four-arm approaches while preventing density-dependent memory growth. Edges are recomputed at each observation step, so a previously weak relation can become dominant when an actor accelerates. The dynamic graph is not treated as ground-truth causality; it is an inductive bias that routes evidence through physically meaningful channels.

Conflict points are obtained from buffered lane-centerline intersections and then consolidated when several segment pairs describe the same physical crossing. The inD dataset provides diverse naturalistic road-user behavior at urban intersections [18], and interaction-based modeling at intersections further supports reasoning over heterogeneous vehicle behaviors [19]. For merging movements, the first common downstream segment defines the interaction location; for following movements, the leading actor's projected lane coordinate replaces a fixed conflict point.

Priority information is encoded as a cue rather than a deterministic rule. Multi-agent tensor fusion [20] illustrates the value of retaining structured pair context, while dynamic graph models such as Trajectory [21] motivate

time-varying relations. The cue reflects approach hierarchy, stop or yield markings when present, and whether an actor has entered the intersection envelope; observed braking can override nominal priority.

Behavior-aware feature construction

Raw kinematics are differentiated only over valid samples and then smoothed with a short causal filter. The graph-structured recurrent treatment in Trajectory++ [22] motivates separating node state from observation validity. Each agent feature concatenates kinematics, lane-relative Freenet coordinates, distance to the nearest conflict point, and a binary hesitation-zone indicator. Time gaps are clipped to avoid unstable values at very low speed.

Interaction messages aggregate edge-conditioned neighbor states. Social-STGCNN [23] demonstrates that graph structure can replace indiscriminate social pooling. Here the message gate increases only when both paths share a conflict point and the estimated arrival gap is small. A residual self-feature protects the actor's own kinematics from being overwritten in crowded scenes, while the original history remains available through a skip connection.

The hesitation-zone indicator is derived from path distance rather than a circular intersection boundary. GroupNet motivates preserving higher-order interaction context beyond isolated pairs, so the two strongest incoming relations are pooled separately from the remaining neighborhood. Within the zone, speed variance, brake persistence, and creep distance distinguish ordinary curvature-related slowing from repeated gap testing.

Observation validity enters both feature normalization and message computation. A temporarily occluded neighbor retains its last valid lane association, but its edge exposure decays exponentially with the missing duration. After 0.8 s without observation, the relation is removed unless map geometry indicates that the actor must still occupy the conflict corridor. This rule prevents stale high-confidence interactions while avoiding abrupt deletion during brief occlusions behind parked vehicles or roadside objects.

Lancin-Mamba Architecture

Lane topology and actor-map fusion

The map encoder divides each lane centerline into short directed segments and initializes each node with local geometry and semantics. Multi-scale graph convolution propagates information along predecessor, successor, left, and right relations. The receptive field therefore expands along reachable routes without mixing unrelated lanes across the intersection. Actor features are projected onto nearby segments using distance, heading agreement, and along-lane progress. Segment-to-actor feedback then summarizes feasible continuations and upcoming conflict geometry.

$$l_s' = \varphi\left(W_0 l_s + \sum_{r\in R} \sum_{u\in N_r(s)} W_r l_u\right) \quad \text{Eq.(5)}$$

The lane update aggregates the current segment feature with neighbors under predecessor, successor, left, and right relation types. Relation-specific matrices preserve directionality, the self-matrix carries the original segment, and φ denotes normalization followed by a gated nonlinear projection.

A topology gate reduces lateral messages when the map indicates a solid boundary or incompatible turn. This preserves the identity of influential scene elements rather than averaging every nearby segment. The actor-map block is repeated twice; deeper stacks provided little gain and increased smoothing in preliminary tests.

Actor-to-lane assignment keeps up to eight segments within 12 m and rejects candidates whose heading differs by more than 75 degrees, except when the actor speed is below 1 m/s. Relation-aware forecasting and learned multi-agent simulation both suggest that scene structure should remain available while behavior evolves. Attention weights therefore combine lateral distance, heading agreement, and progress continuity from the previous sample.

Map caching stores static segment embeddings after the first lane-to-lane propagation. The hierarchical local-global decomposition used by Hit supports this separation of reusable topology from dynamic interaction.

Scene-specific messages update only the subgraph within four hops of an associated segment. Figure 2 shows the resulting topology, temporal-behavior, and decoding stages.

Selective temporal-interaction Mamba

For each actor, the observation history is processed in chronological and reverse directions. The forward stream emphasizes how intent develops, while the reverse stream provides a complementary summary anchored at the most recent state. Each selective block generates input-dependent step size, input projection, and output projection parameters. When a sample contains little new information, the recurrent state changes slowly; abrupt braking or entry into a conflict zone produces a larger update.

$$m_i^t = \sum_{j \in N(i)} \alpha_{ij}^t W_m h_j^t \, , \alpha_{ij}^t = \frac{e_{ij}^t}{\sum_{r \in N(i)} e_{ir}^t} \quad \text{Eq.(6)}$$

The interaction message is a normalized weighted sum of transformed neighbor states. Normalization keeps its magnitude stable as density changes. Only the six neighbors with the highest exposure scores are included, and a residual connection retains the target actor's own state.

$$s_t = A_t s_{t-1} + B_t z_t, o_t = C_t s_t + D z_t \quad \text{Eq.(7)}$$

The selective state-space recurrence updates latent state s_t from input z_t. The discretized transition, input projection, and output projection depend on the current token, allowing the block to preserve informative behavior cues and suppress routine samples. The D term provides a direct input-to-output path.

Behavior-graph messages are injected before each state update rather than appended only at the final time. This allows the temporal state to encode a response sequence: another actor brakes, the target maintains speed, and confidence in the passing mode rises. Layer normalization and gated residuals stabilize training under variable actor counts. Two bidirectional blocks with 128 hidden channels were sufficient; additional blocks increased latency without improving six-second displacement error.

$$h_i{}^M = W_f [o_i{}^{\rightarrow} \| o_i{}^{\leftarrow}] + h_i{}^0 \quad \text{Eq.(8)}$$

The terminal temporal representation concatenates forward and reverse selective outputs, projects them through W_f, and adds the initial actor feature. The residual term protects recent kinematics, while bidirectionality summarizes both intent development and evidence near the final observation.

Lane features and the terminal temporal state are fused through a learned gate. The gate can favor topology for a vehicle with weak motion evidence or favor behavior history when lane continuation is obvious. A conflict summary pools the two strongest incoming edges and remains separate from the general neighbor aggregation, preserving the identity of the most consequential negotiations.

$$g_i = \sigma(W_g [h_i{}^M \| l_i \| c_i]) \quad \text{Eq.(9)}$$

The fusion gate balances temporal behavior against lane and conflict context. Element-wise interpolation produces the fused actor representation, so individual feature channels can favor motion evidence or topology instead of forcing the entire representation to rely on one source.

Sequences are padded only across actors, not across time, because the observation grid is fixed. Hippo's online polynomial memory and the Linear State-Space Layer formulation motivate propagating a compact state even when an input is masked. Invalid samples therefore cannot enter through the input projection, although the transition continues to carry prior evidence. Forward and reverse streams share neither parameters nor normalization statistics.

The terminal state is transferred to the prediction head without temporal pooling over map tokens. Lane-based proposals and dense goal-set prediction motivate keeping endpoint selection separate from continuous trajectory refinement. This separation also prevents the long-memory block from spending capacity on a large, sparsely occupied endpoint grid.

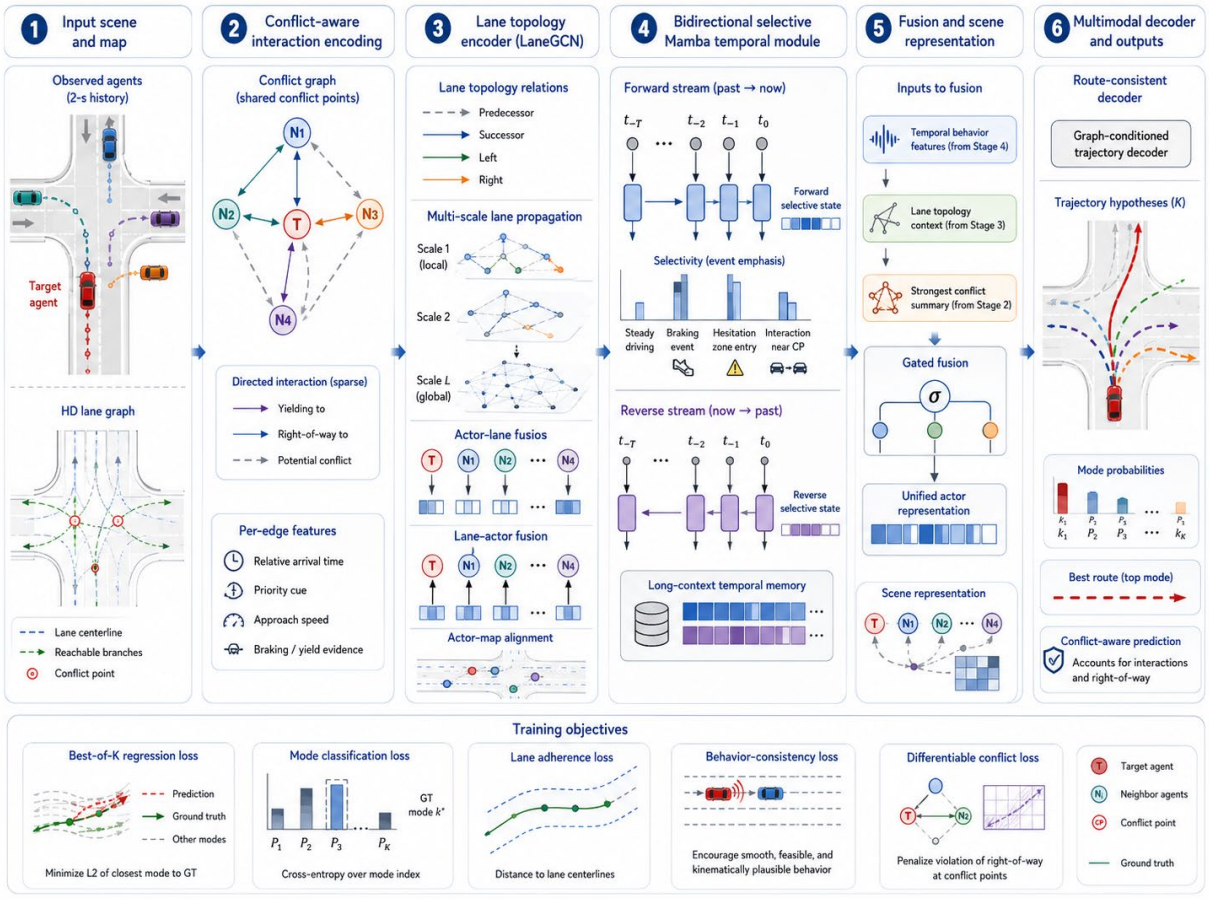

Figure 2. Overall Lancin-Mamba pipeline from lane topology and behavior edges to multimodal trajectories.

Multimodal decoding and training objective

The decoder first predicts K endpoints on reachable lane branches and then generates a smooth offset sequence from each endpoint-conditioned feature. A mode score combines route compatibility, behavioral consistency, and decoder confidence, while non-maximum suppression does not merge modes assigned to different yield/pass relations. This matters when two futures share an initial path but differ in conflict-zone timing.

Training uses a best-of-K regression term, a cross-entropy term for mode selection, a lane adherence term, a behavior-consistency term, and a differentiable conflict penalty. The conflict term evaluates joint occupancy only for graph edges with shared lane conflicts, avoiding global pairwise repulsion. At inference, the model predicts every actor independently conditioned on the shared encoded scene; joint consistency is encouraged by shared interaction features and the training objective rather than combinatorial joint decoding.

$$L = L_{\text{reg}} + \lambda_{\text{cls}} L_{\text{cls}} + \lambda_{\text{lane}} L_{\text{lane}} + \lambda_{\text{beh}} L_{\text{beh}} + \lambda_{\text{conf}} L_{\text{conf}} \quad \text{Eq.(10)}$$

The total objective combines best-of-K trajectory regression, mode classification, lane adherence, behavior consistency, and conflict-point penalties. After warm-up, the classification, lane, behavior, and conflict weights are 1.0, 0.5, 0.25, and 0.20, respectively, keeping geometric accuracy dominant while adding structured constraints.

$$\text{ADE}_k = \frac{1}{T_f} \sum_{\tau=1}^{T_f} \|\hat{\mathbf{y}}_{k,\tau} - \mathbf{y}_\tau\|_2 \,, \text{FDE}_k = \|\hat{\mathbf{y}}_{k,T_f} - \mathbf{y}_{T_f}\|_2 \quad \text{Eq.(11)}$$

ADE measures mean Euclidean displacement over the full prediction horizon, whereas FDE measures endpoint displacement. Reported minADE and minFDE select the best of the six hypotheses for each sample and then average across the test set. Miss rate uses the best final displacement with a 2-m threshold.

https://doi.org/10.64972/jaat.2024v2.326p31e:427-442

Optimization uses Adam with an initial learning rate of 2e-4, cosine decay, gradient clipping at 5.0, and batches containing 32 target-centered scenes. Histories span 2 s at 10 Hz and futures span 6 s. Random rotation, coordinate jitter, agent dropout, and 0.2-s temporal masking improve robustness. The behavior and conflict weights are warmed up over the first eight epochs so that basic kinematics and route alignment stabilize before safety constraints become dominant.

Best-of-K assignment uses final displacement for the first five epochs and a weighted combination of endpoint and full-trajectory error thereafter. The state transition is initialized according to stable diagonal parameterizations studied in DSS and S4D, then optimized jointly with the scene encoder. Mode probabilities use label smoothing of 0.05. The behavior loss compares yield/pass ordering at each conflict point, while the conflict loss integrates overlapping occupancy kernels over time.

Experimental Results

Experimental protocol and implementation

The evaluation is based on a collection of 18,420 carefully chosen benchmark unsignalized-intersection scenarios from urban vehicle trajectories. Four-arm, T-shaped, and skewed crossings are the four types. There are 12,760 scenes for training, 2,710 for validation, and 2,950 for testing in this intersection-disjoint split. GroupNet suggests developing models of pairwise and groupwise relationships at various sizes to enhance the multi-agent forecasting model [24]. In this study, the number of relevant players in a 45-meter conflict neighborhood is used to classify traffic density: sparse situations have no more than three actors, moderate scenes have four to seven, and dense scenes have eight or more. There are 1,004 sparse, 1,126 middling, and 820 dense scenarios in the test set. Each technique predicts six seconds at 10Hz using the same 2-s history and HD-map polylines.

Constant velocity, a vector-map transformer, a recurrent social-pooling predictor, standard LLaMA, LLaMA with temporal attention, and a topology-only LLaMA-Mamba variant are examples of baselines. The links between road users and the surrounding scene environment are explicitly taken into account in relation-aware trajectory forecasting [25]. The validation set's hyperparameters are optimized using the same maximum mode count, K=6. The first two are minimal final displacement error (minFDE) and minimum average displacement error (minADE); the remaining ones are conflict incidence, negative log-likelihood, miss rate at a 2-m final threshold, and predicted calibration error. On an RTX 3060 GPU, runtime is measured with a batch size of one after 200 warm-up frames and averaged over 2,000 frames.

The primary model includes two Lancin fusion rounds, two bidirectional selection blocks, six output modes, and 128-dimensional actor, lane, and temporal states. There are 9.7 million parameters in it. Scene-level consistency for interacting actors is the main emphasis of joint multi-agent behavior models [26]. Training takes 6.3 hours on a single GPU and converges after 44 epochs. No test results are taken into account; the selected checkpoint is the one that minimizes a validation score that includes minFDE, miss rate, and conflict incidence.

No physical crossing should be counted in more than one split, and scenes from the same recording must be separated by at least eight seconds in order to avoid sampling bias. The effectiveness of combining local lane layout with global agent interactions is demonstrated by hierarchical vector-map forecasting [27]. The drive is distributed as follows: 43.6% straight, 31.2% left turn, and 25.2% right turn. 48.9% of the target vehicles yield prior to the initial conflict point. For gaps shorter than 0.3 s, position tracks are resampled to 10 Hz using cubic interpolation; for longer gaps, missing-value masks are retained. Polylines should be sampled every 1.5 meters and clipped to a 100-meter square with the target in the center. The preprocessing stages are the same for all of the aforementioned techniques.

Bootstrap confidence intervals produced from 1,000 scene-level resamples are used to display uncertainty. An online recurrent-memory system called HiPPO is used to gradually compress cumulative sequence history [28]. The 95% confidence interval for the difference between the proposed and Lancin minFDE is 0.31 to 0.48 m, whereas the interval for the miss-rate difference is 2.5 to 4.3 percentage points. Since every algorithm predicts the same scenes, paired resampling is used. In order to prevent a few graph-heavy scenes from obscuring the overall throughput, runtime variability is presented as a median and a 95th-percentile value in addition to the mean.

https://doi.org/10.64972/jaat.2024v2.326p31e:427-442

Model construction is preceded by data quality inspection. Recurrent, convolutional, and continuous-time sequence models are connected in a single state-space form by linear state-space layers [29]. 3.7% of the candidate scenes were eliminated due to the removal of tracks with more than 30% missing history, fewer than 20 valid future samples, or a map association farther than 4m. Lane association is accurate in 96.3% of cases and conflict-point construction is accurate in 94.7% of situations when 300 test scenarios are manually examined. The majority of the remaining mistakes are informal curbside walkways or temporary construction markings. Only the suggested model has derived conflict-edge properties; all learning baselines are given the same cleaned tracks and vector maps. The identical target-centered coordinates and observation masks are used in both constant-velocity and social-pooling baselines.

Accuracy, safety, and calibration

Over the whole test set, Constant Velocity obtained 1.87-m minADE and 4.26-m minFDE. Lane segments are interpretable recommendations for multimodal motion forecasting in lane-based trajectory prediction [30]. The aforementioned numbers are reduced to 1.34m and 3.18m through social pooling. Lancin recorded 0.99 m and 2.31 m, whereas the vector-map transformer attained 1.05 m and 2.44 m. Lancin is improved to 0.91-m minADE and 2.12-m minFDE via temporal attention. In comparison to the standard Lancin, the suggested model has achieved 0.82 m and 1.92 m, lowering the error by 17.2% and 16.9%, respectively. Its 10.4% miss rate is 1.7 percentage points lower than Temporal Attention and 3.4 percentage points lower than Lancin.

.

Together, these panels demonstrate that the performance disparity widens once the trajectories hit the conflict region. Figure 3(a) displays horizon-wise minADE, Figure 3(b) displays minFDE for various prediction horizons, and Figure 3(c) compares the 2-m miss rate.

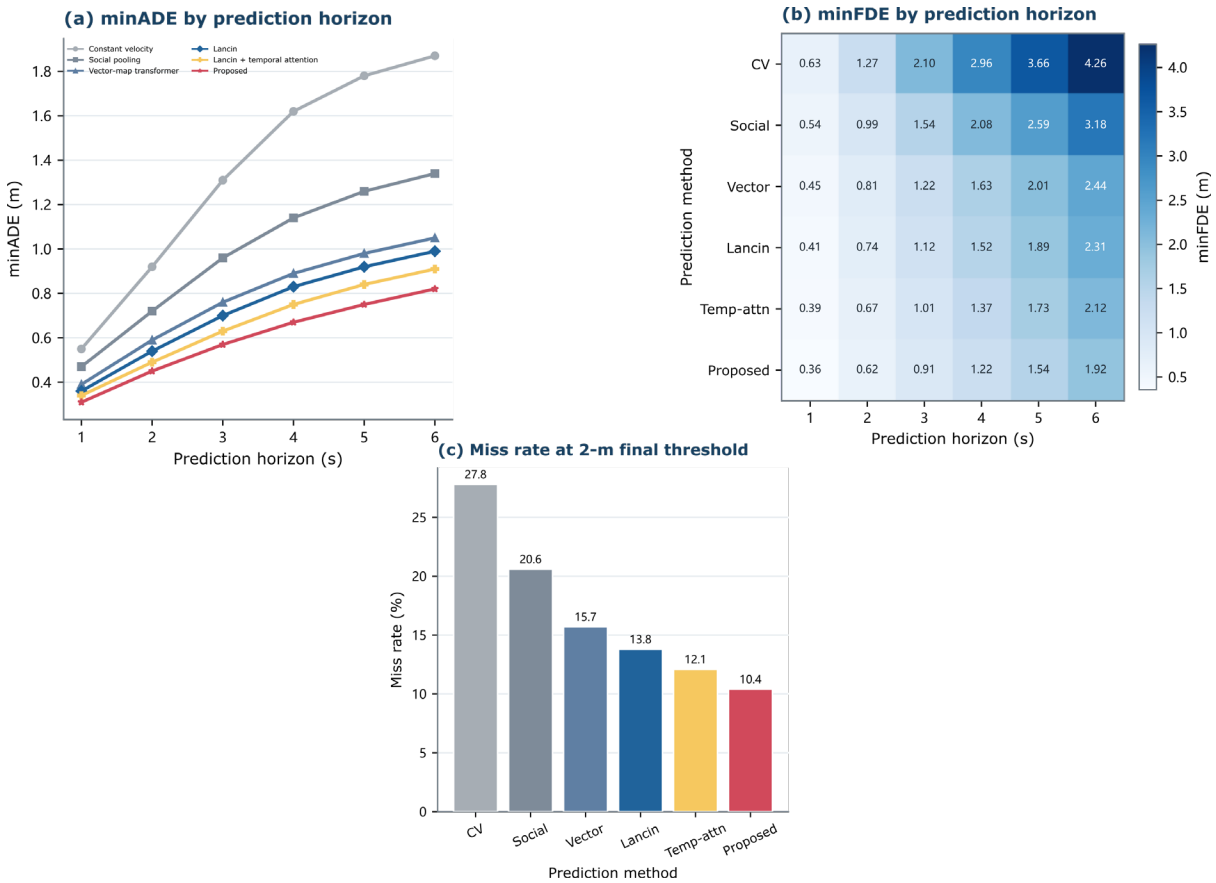

Figure 3. Prediction accuracy by horizon: (a) minADE, (b) minFDE, and (c) miss rate.

In confusing road sceneries, dense objective candidates can provide various trajectories. DenseTNT demonstrates this capability in multimodal forecasting [31]. The strength of the relationship improves the

https://doi.org/10.64972/jaat.2024v2.326p31e:427-442

advantage. The suggested approach recorded a minimal FDE of 2.18m for scenes with the shortest pairwise arrival gap less than 1.0s, while Lancin and temporal attention reported 2.79m and 2.48m, respectively. The suggested approach deviates from temporal attention by as little as 0.08m in a weak-interaction scenario with a time gap greater than 3.0 seconds. Because of greater curve fitting, the advantages are thereby focused in areas where conflict-aware behavior is required rather than being evenly distributed.

For minFDE, the densities of the sparse, moderate, and dense scenes are 1.54, 1.89, and 2.43 meters, respectively. The comparable Lankos values are 1.71, 2.27, and 3.05 meters. Only six interaction neighbors are retained by the model, and the dense-scene improvement is 20.3%. Examining rejected edges reveals that the majority link actors with arrival intervals longer than five seconds or agents to non-conflicting exits. Consequently, the region of simultaneous negotiation coverage for low-value messages will not be considerably reduced by a fixed neighbor budget.

In Figure 4(a), minFDE is stratified by traffic density; in Figure 4(b), the results are grouped by the relative arrival-gap regime; in Figure 4(c), maneuver classes are compared, and the most interaction-sensitive subsets are identified by the dense and small-gap panels.

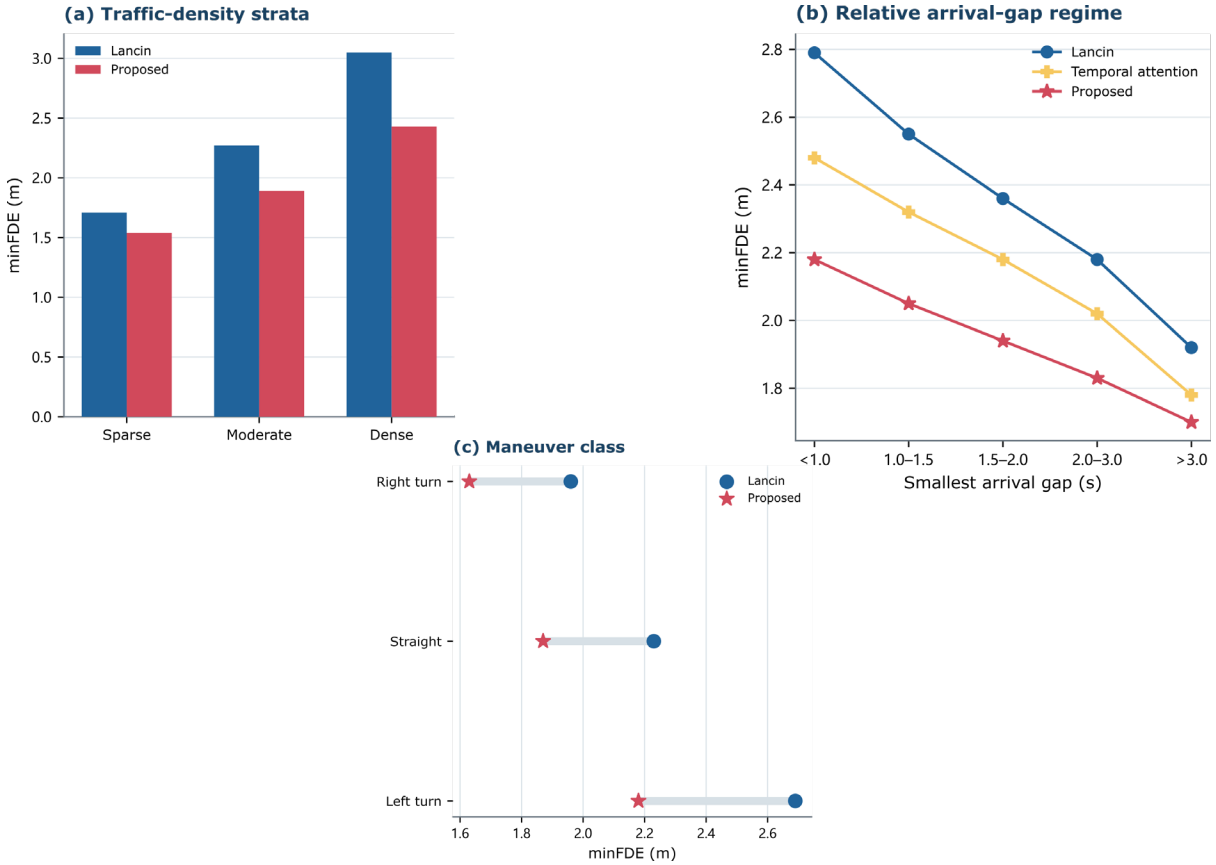

Figure 4. Stratified accuracy: (a) traffic density, (b) relative arrival gap, and (c) maneuver class.

The percentage of anticipated top modes when two actors are within a 2.5-meter radius of the same conflict site for 0.4 seconds is known as the incidence of conflict. Compared to Temporal Attention (3.1%) and Lancin (3.5%), the proposal has a frequency of 2.5%. Compared to Lancenet, it is roughly 28.6% less. Additionally, 2.8% and 1.9% of routes are violated. The aforementioned improvements are not the result of excessive stopping; rather, a heavy collision penalty without behavior conditioning under-progresses by -0.83m, and the mean anticipated progress over six seconds deviates from the ground truth by -0.14m.

Improve accuracy and calibration. The suggested model, Lancin, and temporal attention are projected to have calibration errors of 0.041, 0.066, and 0.052, respectively. The empirical mode success for the highest-confidence quintile is 86.7%, whereas the mean confidence is 88.1%. Since 92.4% of the ground-truth endpoints

are covered by six hypotheses in the lowest-confidence quintile, poor top-mode confidence is linked to actual uncertainty rather than a general model failure.

The route-violation rate is shown in Figure 5(b), the conflict incidence is shown in Figure 5(a), and the confidence reliability curve for safety compatibility, route adherence, and calibration is shown separately in Figure 5(c) without an overall score.

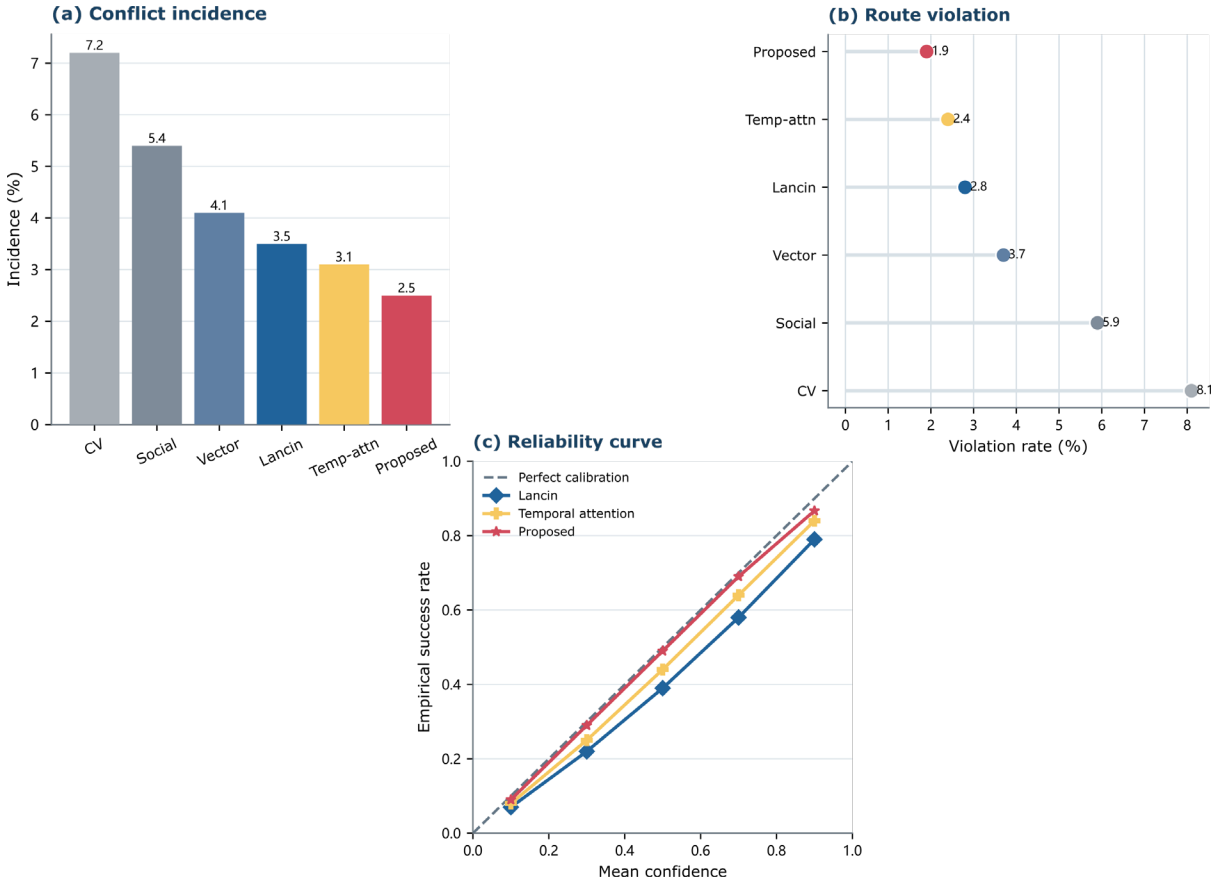

Figure 5. Safety and calibration: (a) conflict incidence, (b) route violation, and (c) reliability curve.

A more thorough distribution of the improvement is provided by endpoint-error distributions. The suggested model has a median final error of 1.21 m, a 75th percentile of 2.04 m, and a 90th percentile of 3.46 m. At the same percentiles, Standard Lancin records 1.43, 2.51, and 4.38 m. The majority of the increase happens above the middle, when competing arrivals and twisting paths create ambiguity; the lower tail just little changes because straightforward straight-through situations are already well predicted. Right turns are the easiest at 1.63m, straight motion is at 1.87m, and left turns are more challenging at 2.18-m minFDE. Nevertheless, the modeling of crossing relations is not more robust, and it is still 0.51m worse than Lancin.

Ablation, robustness, and efficiency

MinFDE rises from 1.92 m to 2.14 m and conflict incidence from 2.5% to 3.2% when conflict-aware interaction edges are eliminated and distance-based neighbors are retained. It is 2.03m and 2.9% without the behavior-consistency loss and 1.96m and 3.4% without the conflict loss. The precision of the interaction graph has improved the most, and the safety level has been raised by the conflict loss. A two-layer gated recurrent unit is employed in place of Mamba to raise minFDE to 2.08m; temporal attention, on the other hand, achieves 1.99m but has a significant latency.

A conflict-incidence graph is shown in Figure 6(b), the changes in minFDE following the removal of a component are shown in Figure 6(a), and the overall calibration changes are shown in Figure 6(c).

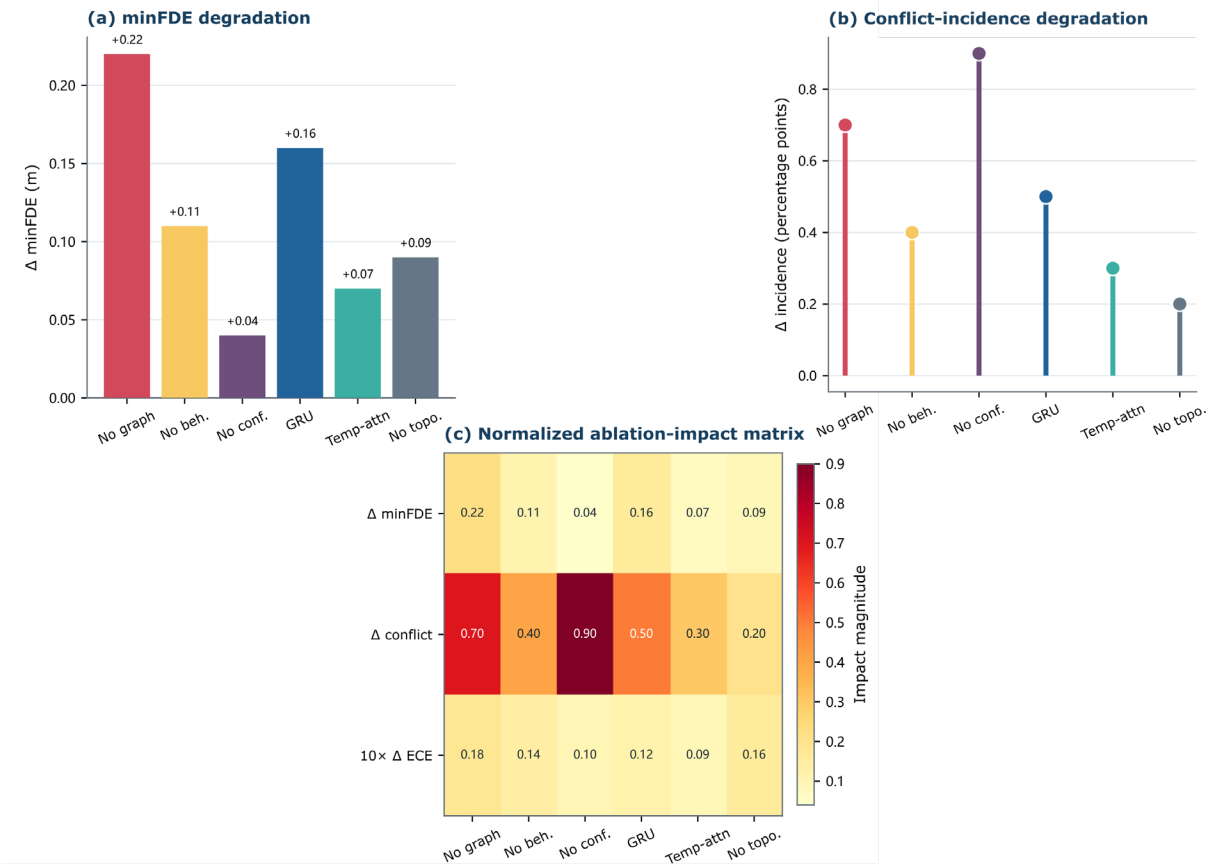

Figure 6. Component ablation: (a) minFDE change, (b) conflict-incidence change, and (c) calibration change.

Another result is the Topology gate. The rate of route violations rises to 2.6% from 1.9% in the absence of boundary- and turn-aware suppression. These errors are concentrated at skewed crossroads where opposing centerlines are within three meters of one another. MinFDE increased by 0.07m when Lancin fusion was reduced from two rounds to one. A shallow map encoder can be utilized with a more potent temporal-interaction module as increasing the number of rounds to four only modifies minFDE by -0.01m and adds 6.8ms.

10%, 20%, and 30% of the historical samples are left out of robustness tests. For long-range sequence challenges, diagonal state-space models can attain the same structured state-space performance with less complicated parameterization [32]. The aforementioned model outperformed temporal-attention Lancin's values of 2.17, 2.39, and 2.71m, achieving minimum FDE values of 1.98, 2.09, and 2.27m. The learnt validity embedding and selected state retention improve by 16.2% at 30% masking. MinFDE grows by 0.13m under 0.25-m Gaussian location noise and by 0.09m under a 5-degree heading perturbation. With a minFDE of 2.68m, the model is still susceptible to extreme occlusion and map misalignment at the same time.

The suggested model, standard Lancin, temporal-attention Lancin, and vector-map transformer have latencies of 31.7 ms, 24.6 ms, 48.9 ms, and 42.3 ms, respectively. Effective Diagonal State-Space Models require careful initialization and parameterization [33]. 1.18GB is the peak GPU memory, and 1.71GB is used for temporal attention. The delay for Mamba and temporal attention increases by 6.4 ms and 17.9 ms, respectively, when the history length is increased from 20 to 40 samples. As a result, the suggested configuration still stays within the 10-Hz planning budget and leaves roughly 68 ms for trajectory optimization, perception, and prediction integration.

The missing-history stress test is displayed in Figure 7(a), latency versus history length is compared in Figure 7(b), peak GPU memory is reported in Figure 7(c), and the history-length panel displays the linear temporal behavior of the selective state-space block.

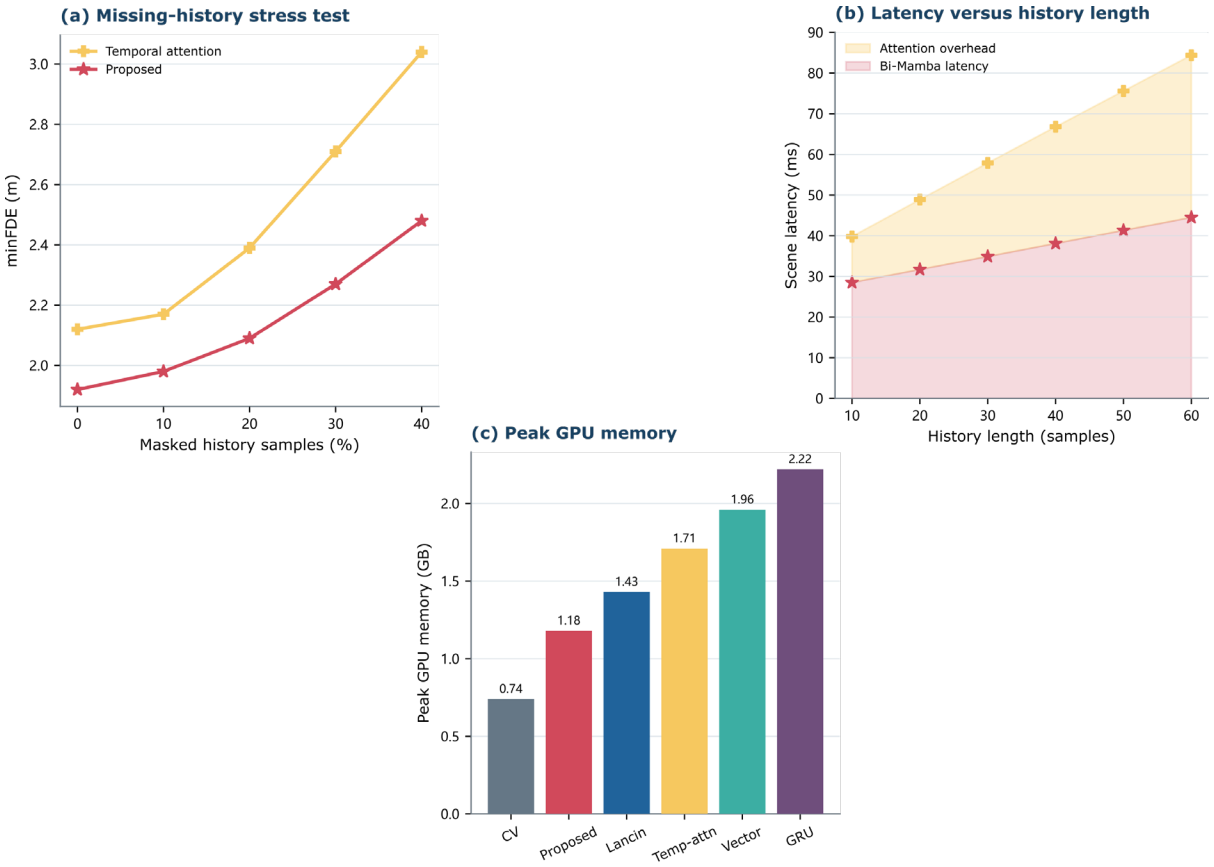

Figure 7. Robustness and efficiency: (a) missing-history stress test, (b) latency versus history length, and (c) peak GPU memory.

Three patterns have been consistently identified via failure analysis. First, lane reachability may be invalid if a car signals late and crosses an unmodeled informal way. Second, braking in the car that the interaction encoder mistakenly attributes to another vehicle may be caused by cyclists or pedestrians outside the vehicle lane graph. Third, a protracted standoff between the two drivers could result in futures that are multimodal in time rather than route. These groups make up roughly 21.8%, 17.6%, and 14.0% of the 307 missing test cases, respectively. They do not alter the observed advantage in vehicle-vehicle negotiating scenarios, but they do result in more complex vulnerable-road-user relations and explicit duration uncertainty.

Mode-count sensitivity demonstrates that an excessive hypothesis budget was not used to obtain the observed improvement. For three, six, and nine modes, the corresponding latencies are 29.6, 31.7, and 35.8 ms, while the MinFDE is 2.08, 1.92, and 1.89 m, respectively. After six modes, coverage saturates, and the remaining three hypotheses primarily replicate previously covered endpoint locations. Consequently, the optimal accuracy-efficiency point is determined to be the six-mode arrangement. The majority of the calibration improvement happens in the interaction-conditioned mode scores rather than after-processing since temperature scaling in the validation split decreased the expected calibration error from 0.044 to 0.041 without altering displacement metrics.

Whether the top-ranked mode fails for the right reasons is ascertained by a qualitative evaluation. Among the five remaining hypotheses in the 200 challenging small-gap scenarios, 61.5% of the top-mode errors still include the ground-truth route, suggesting a timing or confidence problem rather than route omission. A further 22.0% choose the incorrect yield/pass order, 9.5% select the incorrect lane branch, and 7.0% are caused by tracking noise. In 29.5% of the identical circumstances, temporal-attention Lancin generates an inappropriate interaction order. The decreased value of 22.0% suggests that negotiation modulization, rather than a rise in the effective map receptive field, was the primary cause of the increase.

Conclusion

An interaction-behavior-driven Lancin-Mamba model for trajectory prediction at unsupervised crossings is presented in this study. The design allocates negotiation structure to a dynamic conflict-aware graph, long behavioral history to chosen state-space blocks, and lane topology to graph convolution. A conflict-point loss prevents jointly incompatible futures without using global repulsion, and a multimodal decoder combines route feasibility, mode confidence, and behavior consistency.

When compared to regular Lancin, the controlled trial demonstrated a 3.4 percentage point decrease in miss rate and a 16.9% decrease in six-second miniFDE. The greatest gains were shown in dense areas and interactions with arrival gaps of less than one second. Improved calibration and a 28.6% decrease in conflict incidents; scene latency is still 31.7 ms. In addition to offering an engineering rationale for the overall improvement, ablation studies have distinguished the functions of interaction edges, temporal modeling, topology gating, and safety loss.

The model only indirectly depicts non-vehicle influences and is still dependent on lane-map accuracy. Informal routes and long-dead deadlocks are also challenging. Future research will focus on joint prediction-planning assessment, map-uncertainty propagation, and heterogeneous road users. According to the test scope, the results indicate that a model that can detect conflicts, monitor the development of yielding evidence, and identify which lane continuations are physically viable in a unified representation is necessary for high-fidelity unsignalized-intersection prediction.

Author Contributions

Oliver Jõgi contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision. Taavi Rannik contributes to data collection, draft preparation. Egert Võsa contributes to data collection. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

Reference

[1] Rhinehart, N., McAllister, R., Kitani, K. M., & Levine, S. (2019). PRECOG: Prediction conditioned on goals in visual multi-agent settings. Proceedings of the IEEE/CVF International Conference on Computer Vision, 2821–2830. https://doi.org/10.1109/ICCV.2019.00291

[2] Liang, M., Yang, B., Hu, R., Chen, Y., Liao, R., Feng, S., & Urtasun, R. (2020). Learning lane graph representations for motion forecasting. Computer Vision – ECCV 2020, 541–556. https://doi.org/10.1007/978-3-030-58536-5_32

[3] Bonassi, F., Andersson, C., Mattsson, P., & Schön, T. B. (2024). Structured state-space models are deep Wiener models. IFAC-PapersOnLine, 58(15), 247–252. https://doi.org/10.1016/j.ifacol.2024.08.536

[4] Sun, Q., Huang, X., Gu, J., Williams, B. C., & Zhao, H. (2022). M2I: From factored marginal trajectory prediction to interactive prediction. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 6533–6542. https://doi.org/10.1109/CVPR52688.2022.00643

[5] Ivanovic, B., & Pavone, M. (2022). Injecting planning-awareness into prediction and detection evaluation. 2022 IEEE Intelligent Vehicles Symposium (IV), 821–828. https://doi.org/10.1109/IV51971.2022.9827101

[6] Ettinger, S., Cheng, S., Caine, B., Liu, C., Zhao, H., Pradhan, S., Chai, Y., Sapp, B., Qi, C. R., Zhou, Y., Yang, Z., Chouard, A., Sun, P., Ngiam, J., Vasudevan, V., McCauley, A., Shlens, J., & Anguelov, D. (2021). Large scale interactive motion forecasting for autonomous driving: The Waymo Open Motion Dataset. Proceedings of the IEEE/CVF International Conference on Computer Vision, 9710–9719. https://doi.org/10.1109/ICCV48922.2021.00957

[7] Gao, J., Sun, C., Zhao, H., Shen, Y., Anguelov, D., Li, C., & Schmid, C. (2020). VectorNet: Encoding HD maps and agent dynamics from vectorized representation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 11525–11533. https://doi.org/10.1109/CVPR42600.2020.01154

[8] Zeng, W., Liang, M., Liao, R., & Urtasun, R. (2021). LaneRCNN: Distributed representations for graph-centric motion forecasting. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 532–539. https://doi.org/10.1109/IROS51168.2021.9636035

[9] Varadarajan, B., Hefny, A., Srivastava, A., Refaat, K. S., Nayakanti, N., Cornman, A., Chen, K., Douillard, B., Lam, C.-P., Anguelov, D., & Sapp, B. (2022). MultiPath++: Efficient information fusion and trajectory aggregation for behavior prediction. 2022 IEEE International Conference on Robotics and Automation (ICRA), 7814–7821. https://doi.org/10.1109/ICRA46639.2022.9812107

[10] Phan-Minh, T., Grigore, E. C., Boulton, F. A., Beijbom, O., & Wolff, E. M. (2020). CoverNet: Multimodal behavior prediction using trajectory sets. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 14074–14083. https://doi.org/10.1109/CVPR42600.2020.01408

[11] Dendorfer, P., Ošep, A., & Leal-Taixé, L. (2020). Goal-GAN: Multimodal trajectory prediction based on goal position estimation. Computer Vision – ACCV 2020, 405–420. https://doi.org/10.1007/978-3-030-69532-3_25

[12] Yuan, Y., Weng, X., Ou, Y., & Kitani, K. M. (2021). AgentFormer: Agent-aware transformers for socio-temporal multi-agent forecasting. Proceedings of the IEEE/CVF International Conference on Computer Vision, 9813–9823. https://doi.org/10.1109/ICCV48922.2021.00967

[13] Sriram, N. N., Liu, B., Pittaluga, F., & Chandraker, M. (2020). SMART: Simultaneous multi-agent recurrent trajectory prediction. Computer Vision – ECCV 2020, 463–479. https://doi.org/10.1007/978-3-030-58583-9_28

[14] Nayakanti, N., Al-Rfou, R., Zhou, A., Goel, K., Refaat, K. S., & Sapp, B. (2023). Wayformer: Motion forecasting via simple and efficient attention networks. 2023 IEEE International Conference on Robotics and Automation (ICRA), 2980–2987. https://doi.org/10.1109/ICRA48891.2023.10160609

[15] Shi, S., Jiang, L., Dai, D., & Schiele, B. (2022). Motion Transformer with global intention localization and local movement refinement. Advances in Neural Information Processing Systems, 35, 6531–6543. https://doi.org/10.52202/068431-0473

[16] Chang, M.-F., Lambert, J., Sangkloy, P., Singh, J., Bak, S., Hartnett, A., Wang, D., Carr, P., Lucey, S., Ramanan, D., & Hays, J. (2019). Argoverse: 3D tracking and forecasting with rich maps. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8748–8757. https://doi.org/10.1109/CVPR.2019.00895

[17] Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., & Beijbom, O. (2020). nuScenes: A multimodal dataset for autonomous driving. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 11621–11631. https://doi.org/10.1109/CVPR42600.2020.01164

[18] Bock, J., Krajewski, R., Moers, T., Runde, S., Vater, L., & Eckstein, L. (2020). The inD dataset: A drone dataset of naturalistic road user trajectories at German intersections. 2020 IEEE Intelligent Vehicles Symposium (IV), 1929–1934. https://doi.org/10.1109/IV47402.2020.9304839

[19] Roy, D., Ishizaka, T., Mohan, C. K., & Fukuda, A. (2019). Vehicle trajectory prediction at intersections using interaction-based generative adversarial networks. 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2318–2323. https://doi.org/10.1109/ITSC.2019.8916927

[20] Zhao, T., Xu, Y., Monfort, M., Choi, W., Baker, C., Zhao, Y., Wang, Y., & Wu, Y. N. (2019). Multi-agent tensor fusion for contextual trajectory prediction. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 12126–12134. https://doi.org/10.1109/CVPR.2019.01240

[21] Ivanovic, B., & Pavone, M. (2019). The Trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs. Proceedings of the IEEE/CVF International Conference on Computer Vision, 2375–2384. https://doi.org/10.1109/ICCV.2019.00246

[22] Salzmann, T., Ivanovic, B., Chakravarty, P., & Pavone, M. (2020). Trajectron++: Dynamically feasible trajectory forecasting with heterogeneous data. Computer Vision – ECCV 2020, 683–700. https://doi.org/10.1007/978-3-030-58523-5_40

[23] Mohamed, A., Qian, K., Elhoseiny, M., & Claudel, C. (2020). Social-STGCNN: A social spatio-temporal graph convolutional neural network for human trajectory prediction. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 14424–14432. https://doi.org/10.1109/CVPR42600.2020.01443

[24] Xu, C., Li, M., Ni, Z., Zhang, Y., & Chen, S. (2022). GroupNet: Multiscale hypergraph neural networks for trajectory prediction with relational reasoning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 6498–6507. https://doi.org/10.1109/CVPR52688.2022.00639

[25] Choi, C., & Dariush, B. (2019). Looking to relations for future trajectory forecast. Proceedings of the IEEE/CVF International Conference on Computer Vision, 921–930. https://doi.org/10.1109/ICCV.2019.00101

[26] Suo, S., Regalado, S., Casas, S., & Urtasun, R. (2021). TrafficSim: Learning to simulate realistic multi-agent behaviors. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 10400–10409. https://doi.org/10.1109/CVPR46437.2021.01026

[27] Zhou, Z., Ye, L., Wang, J., Wu, K., & Lu, K. (2022). HiVT: Hierarchical vector transformer for multi-agent motion prediction. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8823–8833. https://doi.org/10.1109/CVPR52688.2022.00862

[28] Gu, A., Dao, T., Ermon, S., Rudra, A., & Ré, C. (2020). HiPPO: Recurrent memory with optimal polynomial projections. Advances in Neural Information Processing Systems, 33. https://doi.org/10.5555/3495724.3495849

[29] Gu, A., Johnson, I., Goel, K., Saab, K., Dao, T., Rudra, A., & Ré, C. (2021). Combining recurrent, convolutional, and continuous-time models with linear state-space layers. Advances in Neural Information Processing Systems, 34, 572–585. https://doi.org/10.5555/3540261.3540305

[30] Wang, J., Ye, T., Gu, Z., & Chen, J. (2022). LTP: Lane-based trajectory prediction for autonomous driving. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 17134–17142. https://doi.org/10.1109/CVPR52688.2022.01662

[31] Gu, J., Sun, C., & Zhao, H. (2021). DenseTNT: End-to-end trajectory prediction from dense goal sets. Proceedings of the IEEE/CVF International Conference on Computer Vision, 15303–15312. https://doi.org/10.1109/ICCV48922.2021.01502

[32] Gupta, A., Gu, A., & Berant, J. (2022). Diagonal state spaces are as effective as structured state spaces. Advances in Neural Information Processing Systems, 35, 22982–22994. https://doi.org/10.5555/3600270.3601940

[33] Gu, A., Goel, K., Gupta, A., & Ré, C. (2022). On the parameterization and initialization of diagonal state space models. Advances in Neural Information Processing Systems, 35, 35971–35983. https://doi.org/10.52202/068431-2607