TRAFFUSIONGEN: PARALLEL FUSION FOR TRAFFIC SCENE GENERATION AND COMPLETION

Anonymous authors
Paper under double-blind review

ABSTRACT

Traffic scenario generation requires both scene-level consistency with road geometry and compatibility among interacting participants. Partial observations make this difficult: a model must infer missing traffic states from the remaining evidence. We present TraffusionGen, a framework that treats generation and completion as task-conditioned scene inference. Agents, lanes, and semantic map elements are encoded as shared scene tokens. A Parallel MCG-Attention Block reads the same tokens through two branches: multi-context gating broadcasts shared scene context, while self-attention computes query-specific relational updates. Learnable residual fusion combines these updates before a context-conditioned autoregressive decoder predicts the requested variables. In open-loop experiments on a processed subset of the Waymo Open Motion Dataset, TraffusionGen achieves 0.31 m ADE, a 46.6% reduction relative to the strongest baseline, alongside lower FDE, position MMD, and collision rate than TrafficGen, UniGen, and SceneGen. Its ADE also increases more gradually as vehicle observations are removed. Component ablations support combining both branches and their fusion, highlighting the value of shared context and token-specific interaction for traffic scene inference.

1 INTRODUCTION

Generating a traffic scene requires deciding how participants fit into a road layout and how their motions relate to one another. Completing a partially observed scene poses the same problem with additional evidence: visible participants constrain what can plausibly fill the missing content. Both tasks arise in constructing traffic scenarios for autonomous-driving simulation (Suo et al., 2021; Gulino et al., 2023). We study them as conditional scene inference on the Waymo Open Motion Dataset (WOMD) (Ettinger et al., 2021), with open-loop evaluation of generation and vehicle-masked completion.

Learned traffic models generate scene layouts (Tan et al., 2021), multi-agent rollouts (Suo et al., 2021), or initial states together with trajectories (Feng et al., 2022; Mahjourian et al., 2024). Across these settings, the available context contains heterogeneous information. Road geometry and traffic controls describe the setting in which motion occurs; observed agents provide evidence about its current use. A generator must combine this information when producing new participants or reconstructing missing ones. The representation therefore needs to connect scene-wide information with relationships involving particular agents and map elements.

Consider completing a scene near an intersection. Lane geometry and traffic controls inform the possible paths of a missing participant, while the positions and motions of visible vehicles provide evidence about its relation to the surrounding traffic. These sources have different roles: some information is useful across many participants, whereas other information depends on the particular participant being inferred. Removing vehicle observations changes the relational evidence available to the model while leaving the road context in place. The resulting design question is how to combine shared scene context and participant-specific relationships as the available agent evidence changes.

We combine these two forms of reasoning through parallel branches. Multi-context gating (MCG) pools the scene into a summary and broadcasts a feature-wise transformation to the tokens. Self-attention computes a different aggregation for each query token, retaining relationships among individual scene elements. Both branches read the same incoming representation and produce separate

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

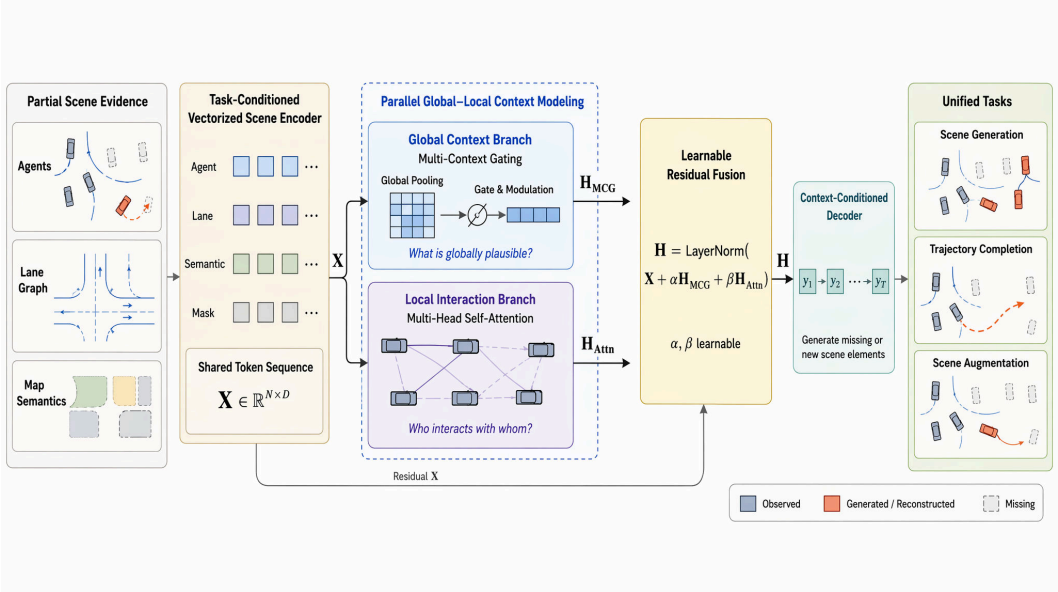


Figure 1: **TraffusionGen overview.** Agent, lane, and semantic tokens feed parallel MCG and self-attention branches. Residual fusion combines their updates before task-conditioned decoding. Different observed and target variables specify generation, completion, or the illustrated extension to augmentation.

updates before fusion. The design gives the decoder access to features shaped by shared context and query-specific interaction within one token representation.

We instantiate this idea in **TraffusionGen**, a framework for scene generation and masked trajectory completion. Dynamic-agent histories, lane polylines, and semantic map elements are projected into shared scene tokens. A stack of **Parallel MCG-Attention Blocks** refines these tokens through residual fusion, and a context-conditioned autoregressive decoder predicts the requested variables. Generation and completion use different observation masks over the same backbone.

1.1 Problem Statement and Contributions

Let $\mathcal{S}$ denote a traffic scene containing dynamic agents, lane geometry, and semantic traffic-control elements. A task specification τ partitions the variables of $\mathcal{S}$ into observed evidence $\mathcal{X}_{\text{obs}}$ and target variables $\mathcal{Y}_{\text{tar}}$. The targets may represent a complete set of traffic participants, missing states from partially observed tracks, or additional agents introduced into an existing scene. We seek a single conditional model

$$p_\theta(\mathcal{Y}_{\text{tar}} \mid \mathcal{X}_{\text{obs}}, \tau), \tag{1}$$

that changes its outputs according to the task while reusing the same representation and reasoning backbone. Under this view, standard generation corresponds to observing the map while treating most agent variables as targets; vehicle-masked completion keeps the visible scene fixed and predicts only the removed agent information. Other masks can be expressed by changing the partition rather than redesigning the network.

Our contributions are threefold:

- **Parallel global-local context fusion.** We introduce a block in which MCG and self-attention operate independently on the same incoming tokens, then contribute learnable residual updates combining scene-level context and token-level relationships.
- **A shared framework for generation and completion.** Vectorized agent, lane, and semantic tokens connect the fusion blocks to a conditional autoregressive decoder. The observed and target variables specify the task within this common backbone.

- **Evaluation across generation and missing observations.** On a processed WOMD subset, we report four generation metrics, completion ADE under increasing vehicle masking, and component ablations. The full model improves on the compared baselines and on variants that remove either context branch.

2 Related Work

2.1 Traffic Simulation and Data-Driven Generation

Rule-based platforms such as SUMO (Lopez et al., 2018), CARLA (Dosovitskiy et al., 2017), SMARTS (Zhou et al., 2021), and HighwayEnv (Leurent, 2018) provide controllable, repeatable scenarios, but hand-written behaviors are difficult to scale to long-tailed interactions. Data-driven simulators instead learn from recorded motion: TrafficSim (Suo et al., 2021) models multi-agent rollouts, SceneGen (Tan et al., 2021) generates scene configurations, TrafficGen (Feng et al., 2022) unifies generation and completion, and UniGen (Mahjourian et al., 2024) jointly models initial states and trajectories. Reliable completion under missing evidence still requires a coherent scene interpretation rather than simple extrapolation of visible tracks.

2.2 Scene-Centric Motion Representation

Large-scale datasets such as WOMD (Ettinger et al., 2021), Argoverse (Chang et al., 2019), and nuScenes (Caesar et al., 2020) have driven rapid progress in multi-agent motion prediction. Social-GAN (Gupta et al., 2018) and Trajectron++ (Salzmann et al., 2020) model multimodal futures, while VectorNet (Gao et al., 2020) and TNT (Zhao et al., 2021) encode map and agent structure using vectorized or graph-based representations. Scene Transformer (Ngiam et al., 2022) reasons jointly across agents, road elements, and time through self-attention. These predictors provide useful representation principles, especially scene-centric tokenization and joint interaction modeling. Our setting differs in that the model must also generate missing scene content and therefore cannot assume that all relevant agents or histories are present at the input.

2.3 Global Context and Local Interaction

Residual connections (He et al., 2016) and self-attention (Vaswani et al., 2017) provide general building blocks for representation learning. MultiPath++ uses multi-context gating to fuse heterogeneous motion features (Varadarajan et al., 2022). TraffusionGen combines MCG with token-level self-attention in independent branches reading the same input before residual fusion. Its architectural focus is this composition within a task-conditioned generator.

Recent work also unifies traffic modeling tasks. SceneDiffuser combines scene initialization and controllable closed-loop rollout through diffusion (Jiang et al., 2024); UniMTD unifies trajectory prediction, imputation, and recovery under missing observations (Yang et al., 2025). These studies motivate shared models across observation patterns. TraffusionGen focuses on the context representation within such a model: parallel MCG and attention branches combine shared scene information with token-level relationships before autoregressive decoding.

3 Method

3.1 Overview: Scene Inference from Partial Evidence

Let $\mathcal{S} = (\mathcal{A}, \mathcal{L}, \mathcal{M})$ contain dynamic agents, lane polylines and connectivity, and semantic map elements. The task indicator τ partitions these variables into observed evidence and targets as in Eq. (1). TraffusionGen has three stages (Fig. 1): modality-specific encoding, parallel context fusion, and autoregressive decoding. The same backbone processes different observation patterns; the evidence mask and requested targets specify the task.

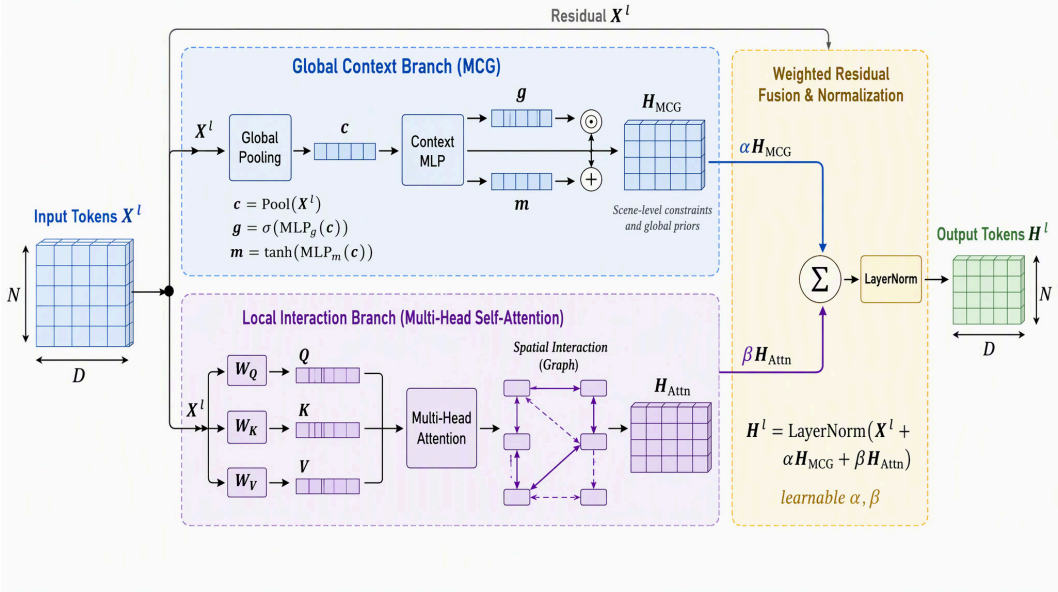


Figure 2: **Parallel MCG-Attention Block.** Both branches receive the same tokens. MCG broadcasts a scene-level gate and modulation, while multi-head self-attention models token-level relationships. Learnable residual fusion adds both updates to the incoming state before normalization.

3.2 Task-Conditioned Vectorized Scene Representation

Traffic scenes combine quantities with very different structure: agent motion is temporal, lane geometry is polyline based, and map semantics are categorical. We use modality-specific encoders to map these heterogeneous inputs into a common D-dimensional feature space, where the fusion blocks model interactions among scene elements.

Dynamic-agent tokens. For agent i at time t, we use the kinematic state

$$\mathbf{s}_i^t = [x_i^t, y_i^t, v_{x,i}^t, v_{y,i}^t, \psi_i^t, a_i^t, \omega_i^t] \in \mathbb{R}^7, \tag{2}$$

which records position, velocity, heading, acceleration, and angular velocity. The same agent encoder is shared across participants,

$$\mathbf{h}_{i,t}^A = \mathrm{MLP}_A(\mathbf{s}_i^t) \in \mathbb{R}^D, \tag{3}$$

and the valid observations belonging to one track are aggregated into an agent-level token $\mathbf{h}_i^A$. Sharing the encoder makes the representation permutation equivariant with respect to agent identity: reordering agents changes only the token order, not their meaning.

Lane and semantic tokens. A lane j is represented as an ordered polyline $\mathbf{l}_j$ together with direction and topological attributes. A hierarchical polyline encoder first embeds its points and then pools them into

$$\mathbf{h}_j^L = \mathrm{PolylineEncoder}(\mathbf{l}_j) \in \mathbb{R}^D. \tag{4}$$

Traffic lights, stop lines, lane types, and other symbolic elements are encoded using a semantic embedding with positional information,

$$\mathbf{h}_k^M = \mathrm{EmbedSemantic}(\mathbf{m}_k) \in \mathbb{R}^D. \tag{5}$$

Modality embeddings identify whether a token originates from an agent, lane, or semantic element, while positional information preserves where that element lies in the common scene frame.

The resulting token set is

$$\mathbf{X}^0 = \mathrm{Concat}\Big([\mathbf{h}_i^A]_{i=1}^{N_A}, [\mathbf{h}_j^L]_{j=1}^{N_L}, [\mathbf{h}_k^M]_{k=1}^{N_M}\Big) \in \mathbb{R}^{N \times D}, \tag{6}$$

where $N = N_A + N_L + N_M$. The task specification τ controls which raw attributes contribute to these tokens. Consequently, changing from full generation to masked completion changes the evidence presented to the encoder, but does not change the encoder itself.

The shared feature dimension lets the fusion block operate on agents and infrastructure together, while modality and positional information distinguish their roles. In particular, an agent token can use both other participants and road elements as context in the same attention operation.

3.3 Parallel Global-Local Context Modeling

Global pooling forms a compact scene summary, while self-attention retains token-level resolution. TraffusionGen computes these complementary representations from the same input and combines their updates through residual fusion. Here, "local" refers to token-level relationships; the attention operator is not restricted to spatial neighbors.

For block ℓ, let $\mathbf{X}^{\ell-1} \in \mathbb{R}^{N\times D}$ denote its input. The block computes a global MCG update and an interaction-aware attention update in parallel, as shown in Fig. 2. Multi-context gating follows the context-fusion principle introduced for heterogeneous motion features in MultiPath++ (Varadarajan et al., 2022); our contribution is to couple that global broadcast mechanism with an independent token-level attention path and to fuse the two through a shared residual state.

Global context through MCG. The MCG path provides a common conditioning signal across the scene. It first summarizes the visible token set into a scene descriptor,

$$\mathbf{c}^{\ell} = \mathrm{Pool}\big(\mathbf{X}^{\ell-1}\big) \in \mathbb{R}^{D}. \tag{7}$$

From this descriptor, two lightweight networks predict a gate and a modulation vector,

$$\mathbf{g}^{\ell} = \sigma\Big(\mathrm{MLP}_{g}^{\ell}(\mathbf{c}^{\ell})\Big), \tag{8}$$

$$\mathbf{m}^{\ell} = \tanh\Big(\mathrm{MLP}_{m}^{\ell}(\mathbf{c}^{\ell})\Big). \tag{9}$$

Here, σ is the element-wise sigmoid. Both D-dimensional vectors are broadcast over the N token rows, producing the context-conditioned update

$$\mathbf{H}_{\mathrm{MCG}}^{\ell} = \mathbf{g}^{\ell} \odot \mathbf{X}^{\ell-1} + \mathbf{m}^{\ell}. \tag{10}$$

The gate reweights feature channels according to the scene, while the modulation term broadcasts shared context across agents and map elements. This path communicates a compact scene summary to every token at linear cost in the number of elements.

For token i, Eq. (10) reads $\mathbf{h}_{\mathrm{MCG},i}^{\ell} = \mathbf{g}^{\ell} \odot \mathbf{x}_{i}^{\ell-1} + \mathbf{m}^{\ell}$, where $\odot$ denotes element-wise multiplication. The gate rescales each token's features and the modulation adds a shared offset. Thus, $\mathbf{H}_{\mathrm{MCG}}^{\ell} \in \mathbb{R}^{N\times D}$ retains distinct token features under a common scene-dependent transformation.

Local interaction through self-attention. The parallel attention path preserves token-level resolution. For each head $r \in \{1, \ldots, h\}$, where h is the number of heads, learned projections map the same input into queries, keys, and values,

$$\mathbf{Q}_r = \mathbf{X}^{\ell-1}\mathbf{W}_r^{Q}, \quad \mathbf{K}_r = \mathbf{X}^{\ell-1}\mathbf{W}_r^{K}, \quad \mathbf{V}_r = \mathbf{X}^{\ell-1}\mathbf{W}_r^{V}, \tag{11}$$

then computes attention with query/key dimension d_k and row-wise softmax over the key tokens:

$$\mathrm{head}_r = \mathrm{softmax}\left(\frac{\mathbf{Q}_r\mathbf{K}_r^{\top}}{\sqrt{d_k}}\right)\mathbf{V}_r. \tag{12}$$

The heads are concatenated and projected,

$$\mathbf{H}_{\mathrm{attn}}^{\ell} = \mathrm{Concat}(\mathrm{head}_1, \ldots, \mathrm{head}_h)\mathbf{W}^{O}. \tag{13}$$

Because agents, lanes, and traffic controls are represented in the same token set, attention can model dependencies among agents and between agents and map elements with one relational operator. This scene-centric interaction modeling is consistent with the joint reasoning motivation of attention-based multi-agent predictors (Ngiam et al., 2022).

The row-wise weights in Eq. (12) make this update specific to each query. Two agent tokens can aggregate different lane or participant features even though they share the MCG conditioning signal. Multiple heads provide several such relational aggregations before projection back to dimension D. Both branch outputs consequently retain the $N \times D$ token layout, allowing fusion to combine their updates for each corresponding scene element.

Residual fusion. We combine the branch updates at the token level so that the next block receives both forms of context in a common representation. A residual connection carries the incoming state into this update:

$$\mathbf{X}^{\ell} = \mathrm{LayerNorm}\big(\mathbf{X}^{\ell-1} + \alpha_{\ell}\mathbf{H}^{\ell}_{\mathrm{MCG}} + \beta_{\ell}\mathbf{H}^{\ell}_{\mathrm{attn}}\big)\,, \tag{14}$$

where α_{ℓ} and β_{ℓ} are learnable scales controlling the two branches' contributions. Stacking L blocks yields the context representation $\mathbf{H} = \mathbf{X}^{L}$.

Within a block, neither branch consumes the other's output. Across blocks, both read the previously fused state, so the next pooled summary and the next attention weights can reflect both earlier updates. This separates the computation of the two current updates while allowing their information to interact over depth. The layer-specific coefficients are learned with the model parameters; the gate and attention weights depend on the current scene features.

3.4 Context-Conditioned Autoregressive Decoding

The encoder and fusion stack produce a scene representation, but the meaning and number of outputs depend on the task. We therefore express the requested target as an ordered sequence $\mathcal{Y}_{\mathrm{tar}} = (\mathbf{y}_1, \ldots, \mathbf{y}_K)$. A target unit can denote the state and trajectory of a new agent, a missing portion of an observed track, or another task-specific scene variable. Conditioned on the fused tokens and task indicator, the decoder factorizes the joint distribution as

$$p_{\theta}(\mathcal{Y}_{\mathrm{tar}} \mid \mathcal{X}_{\mathrm{obs}}, \tau) = \prod_{k=1}^{K} p_{\theta}(\mathbf{y}_k \mid \mathbf{y}_{<k}, \mathbf{H}, \tau)\,. \tag{15}$$

At step k, the decoder conditions on the fused scene context and previous targets. These two inputs have distinct roles: $\mathbf{H}$ represents the observed evidence, while $\mathbf{y}_{<k}$ records the target content already generated. Conditioning on the prefix allows subsequent predictions to account for earlier outputs as the requested scene content is constructed.

For **scenario generation**, map and semantic tokens provide context for sequential agent generation. For **trajectory completion**, observed states remain fixed and only masked targets are decoded. The formulation also accommodates **scene augmentation** by treating existing agents as context and additional agents as targets.

3.5 Unified Learning Objective

Training examples are converted into task instances by selecting a task τ and partitioning each scene into $\mathcal{X}_{\mathrm{obs}}$ and $\mathcal{Y}_{\mathrm{tar}}$. We optimize the conditional negative log-likelihood

$$\mathcal{L}_{\mathrm{gen}}(\theta) = -\mathbb{E}_{(\mathcal{S},\tau)}\left[\sum_{k=1}^{K} \log p_{\theta}(\mathbf{y}_k \mid \mathbf{y}_{<k}, \mathbf{H}, \tau)\right]. \tag{16}$$

Only target variables contribute to the loss; observed variables serve as context. Training uses ground-truth prefixes, whereas inference feeds back generated outputs. Task masks expose the shared parameters to different observation patterns. The masking experiments below assess removal of vehicle observations.

4 Experiments

The experiments examine three questions: how the complete model performs in standard generation, how completion accuracy changes as vehicle observations are removed, and how individual components contribute to the full model. Training curves are provided in Appendix A.

4.1 Experimental Setup

Dataset and preprocessing. We conduct experiments on the Waymo Open Motion Dataset (WOMD) (Ettinger et al., 2021), which contains urban and highway scenes with diverse road geometry and multi-agent interaction. Each processed example is cropped to a $120\,\mathrm{m} \times 120\,\mathrm{m}$ region centered on the ego vehicle, and scenes with fewer than eight agents are excluded. The resulting subset contains 50,000 training scenarios, 5,000 validation scenarios, and 1,000 test scenarios, with an average of 12.5 agents per scene (Table 1). These counts describe the processed subset used in this paper rather than the complete WOMD release.

Table 1: Statistics of the processed WOMD subset. Counts refer to processed scenarios rather than the full WOMD release.

Statistic	Value
Training scenarios	50,000
Validation scenarios	5,000
Test scenarios	1,000
Average agents per scene	12.5

Evaluation tasks. We evaluate **standard generation**, conditioned on map and semantic context, and **vehicle-masked completion**, which reconstructs removed agent observations from the visible scene. Increasing the masking ratio reduces available vehicle evidence while retaining road context. Generation is evaluated using the four metrics below; completion is evaluated through ADE across masking ratios. Augmentation is an extension of the formulation rather than an evaluated task in this study.

Baselines and metrics. We compare with SceneGen (Tan et al., 2021), TrafficGen (Feng et al., 2022), and UniGen (Mahjourian et al., 2024). ADE and FDE measure mean and final trajectory deviation, position MMD measures distribution mismatch, and collision rate is a unitless measure of local physical compatibility. All metrics are lower-is-better; together they test whether a method improves several failure modes rather than one metric in isolation.

4.2 Standard Generation

Table 2 is the central comparison. TraffusionGen obtains 0.31 m ADE and 0.40 m FDE, reducing the strongest baseline values by 46.6% and 46.7%. Position MMD decreases from 0.35 to 0.12 (65.7%), and collision rate from 0.38 to 0.18 (52.6%).

Table 2: **Overall comparison on the processed WOMD test subset.** All metrics are lower-is-better. Collision rate is reported as a unitless fraction. The best result is shown in **bold**, and the strongest baseline is underlined.

Method	ADE (m) ↓	FDE (m) ↓	Position MMD ↓	Collision rate ↓
SceneGen	0.82	1.05	0.58	0.55
TrafficGen	<u>0.58</u>	<u>0.75</u>	<u>0.35</u>	0.42
UniGen	0.65	0.82	0.41	<u>0.38</u>
TraffusionGen (Ours)	**0.31**	**0.40**	**0.12**	**0.18**

The metrics assess different aspects of the generated scenes: displacement errors describe agreement with reference trajectories, position MMD describes distribution mismatch, and collision rate reflects interactions between participants. Reading these columns together distinguishes improvements in trajectory accuracy from improvements that also extend to the other reported criteria. The baseline rankings illustrate this distinction: TrafficGen has the lowest baseline ADE, FDE, and position MMD, whereas UniGen has the lowest baseline collision rate. TraffusionGen improves on both profiles, with lower trajectory errors accompanied by lower distribution mismatch and collision rate.

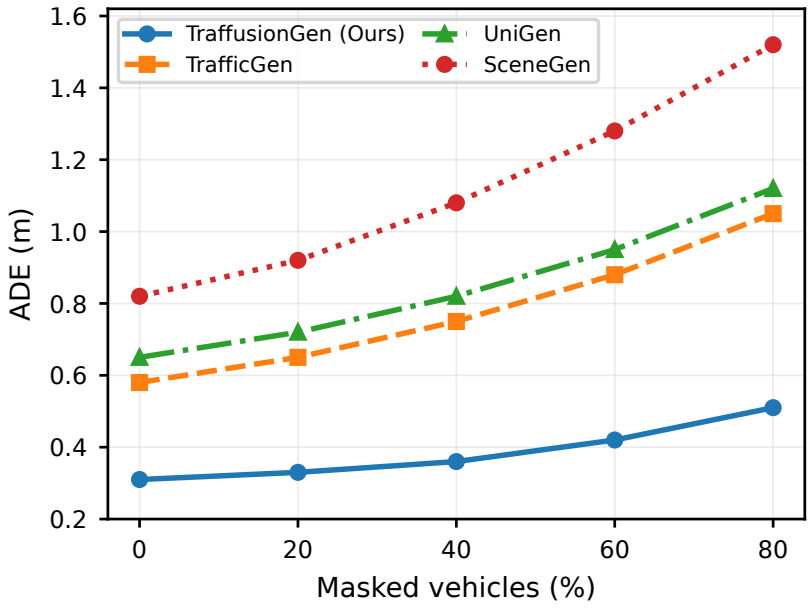


Figure 3: **Robustness to vehicle masking.** ADE as masked vehicles increase from 0% to 80%. TraffusionGen remains below all baselines across the masking range.

Table 3: **Component ablation.** Each variant removes one component from TraffusionGen. Lower is better.

Variant	ADE (m) ↓	Collision rate ↓
w/o MCG path	0.52	0.38
w/o self-attention	0.48	0.35
w/o residual fusion	0.45	0.32
w/o context pooling	0.42	0.30
Full model (Ours)	**0.31**	**0.18**

4.3 Completion under Missing Vehicle Observations

Vehicle masking tests the representation as its input evidence changes. The road context remains available while fewer observed vehicles can inform the reconstruction of missing participants. Varying the masking ratio therefore examines sensitivity to evidence availability across a range of conditions. Figure 3 increases the masked-vehicle fraction from 0% to 80%. TraffusionGen rises from 0.31 m ADE to 0.51 m, whereas TrafficGen, UniGen, and SceneGen reach 1.05, 1.12, and 1.52 m under the same severe mask.

All methods incur greater error as more vehicle observations are removed, but TraffusionGen's ADE increases more gradually. At 80% masking, its error is less than half of TrafficGen's. The widening gap indicates that the model's advantage persists as the visible-agent context becomes sparse, linking the generation comparison to the central problem of inference from incomplete scenes.

4.4 Component Ablation

Table 3 examines two aspects of the design: the branch-specific updates and the operations that form and combine them. Removing MCG increases ADE to 0.52 m and collision rate to 0.38; removing self-attention raises them to 0.48 m and 0.35. Both variants lose performance on both metrics, supporting the inclusion of shared scene modulation alongside query-specific relational aggregation.

The pooling and fusion ablations address how the updates are formed and combined. Pooling constructs the descriptor used by MCG, while residual fusion integrates the branch updates with the incoming state. Removing either operation also increases ADE and collision rate. Thus, the observed gains depend on both the available update paths and their organization within the block, supporting the complete combination used by TraffusionGen.

The two metrics also rank the ablated configurations identically: removing MCG produces the highest errors, followed by removing self-attention, residual fusion, and context pooling. The full model has the lowest values on both metrics. This agreement shows that the relative performance of these configurations is consistent across trajectory accuracy and collision rate; none of the reported variants achieves lower ADE by accepting a higher collision rate than another variant.

5 Discussion and Limitations

Shared context and differentiated interactions. The generation and completion results show the usefulness of the combined representation across different levels of available agent evidence. Component ablations further support retaining both shared modulation and query-specific aggregation. Together, these findings motivate a scene representation that uses common context while maintaining distinct relationships for individual tokens.

Implications for driving world models. Driving world models connect scene generation to future prediction and decision-making. GAIA-1 conditions video generation on video, text, and actions (Hu et al., 2023); Drive-WM generates multiview futures and uses them for planning (Wang et al., 2024). TraffusionGen suggests a complementary direction at the level of structured scene representation. First, shared scene conditioning and token-specific interaction offer a way to organize road context and participant relationships within a common state representation. Second, completion highlights the role of reconstructing missing scene content from available evidence. Our vehicle-masking results motivate exploring how such reconstruction could provide an initial scene representation for a driving world model when participant observations are incomplete. A possible extension would use the resulting agent and map representations as context for action-conditioned future prediction, connecting scene reconstruction with temporal evolution. This would require learning and evaluating the transition from the reconstructed scene under ego actions. The resulting research question is how jointly representing shared context and individual relationships affects the consistency of future rollouts.

Limitations and next steps. The experiments use open-loop evaluation on a processed WOMD subset, examine completion through vehicle masking, and report point estimates. Evaluation across datasets, additional observation patterns, and repeated runs would help characterize generalization and variability. Reactive closed-loop evaluation is a further step toward assessing longer-term interaction with an ego policy.

6 Conclusion

TraffusionGen treats traffic generation and completion as scene inference under partial evidence. Parallel MCG and self-attention branches combine shared scene conditioning with token-specific relational aggregation, followed by residual fusion and autoregressive decoding. On the processed WOMD subset, the model improves the reported generation metrics and maintains lower ADE under vehicle masking; component ablations support retaining both branches and their fusion. The central design insight is to give both forms of context a direct role in constructing scene representations. This also motivates investigating structured scene completion as a starting point for action-conditioned driving world models.

AI USE STATEMENT

During manuscript preparation and revision, we used generative AI tools to help organize, draft, and edit text; identify and summarize related literature; discuss interpretations of the reported results and possible research directions; and clarify notation. AI assistance also supported reference formatting, LaTeX conversion, and scripts for compilation and document checks. This revision did not involve generating new experimental data or running new model experiments. The authors take responsibility for the final text, claims, and supporting materials, including AI-assisted content.

REFERENCES

Holger Caesar et al. nuScenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Ming-Fang Chang et al. Argoverse: 3D tracking and forecasting with rich maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Alexey Dosovitskiy et al. CARLA: An open urban driving simulator. In *Conference on Robot Learning (CoRL)*, 2017.

Scott Ettinger et al. Large scale interactive motion forecasting for autonomous driving: The Waymo open motion dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Lan Feng, Quanyi Li, Zhenghao Peng, Shuhan Tan, and Bolei Zhou. TrafficGen: Learning to generate diverse and realistic traffic scenarios. In *European Conference on Computer Vision (ECCV)*, 2022.

Jiyang Gao et al. VectorNet: Encoding HD maps and agent dynamics from vectorized representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Cole Gulino et al. Waymax: An accelerated, data-driven simulator for large-scale autonomous driving research. In *Advances in Neural Information Processing Systems*, 2023.

Agrim Gupta et al. Social GAN: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Kaiming He et al. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. GAIA-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023.

Chiyu Max Jiang, Yijing Bai, Andre Cornman, Christopher Davis, Xiukun Huang, Hong Jeon, Sakshum Kulshrestha, John Lambert, Shuangyu Li, Xuanyu Zhou, Carlos Fuertes, Chang Yuan, Mingxing Tan, Yin Zhou, and Dragomir Anguelov. SceneDiffuser: Efficient and controllable driving simulation initialization and rollout. In *Advances in Neural Information Processing Systems*, 2024.

Edouard Leurent. An environment for autonomous driving decision-making. GitHub repository: highway-env, 2018.

Pablo Alvarez Lopez et al. Microscopic traffic simulation using SUMO. In *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2018.

Reza Mahjourian, Rongbing Mu, Valerii Likhosherstov, Paul Mougin, Xiukun Huang, Joao Messias, and Shimon Whiteson. UniGen: Unified modeling of initial agent states and trajectories for generating autonomous driving scenarios. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 16367–16373, 2024. doi: 10.1109/ICRA57147.2024.10611115.

Jiquan Ngiam et al. Scene transformer: A unified architecture for predicting multiple agent trajectories. In *International Conference on Learning Representations (ICLR)*, 2022.

Tim Salzmann et al. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *European Conference on Computer Vision (ECCV)*, 2020.

Simon Suo et al. TrafficSim: Learning to simulate realistic multi-agent behaviors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Shuhan Tan et al. SceneGen: Learning to generate realistic traffic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Balakrishnan Varadarajan, Ahmed Hefny, Avikalp Srivastava, Khaled S. Refaat, Nigamaa Nayakanti, Andre Cornman, Kan Chen, Bertrand Douillard, Chi-Pang Lam, Dragomir Anguelov, and Benjamin Sapp. MultiPath++: Efficient information fusion and trajectory aggregation for behavior prediction. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

Ashish Vaswani et al. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Yuqi Wang, Jiawei He, Lue Fan, Hongxin Li, Yuntao Chen, and Zhaoxiang Zhang. Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14749–14759, 2024.

Songru Yang, Zhenwei Shi, and Zhengxia Zou. Unified multi-agent trajectory modeling with masked trajectory diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 27563–27574, 2025.

Hang Zhao et al. TNT: Target-driven trajectory prediction. In *Conference on Robot Learning (CoRL)*, 2021.

Ming Zhou et al. SMARTS: An open-source scalable multi-agent RL training school for autonomous driving. In *Proceedings of the 2020 Conference on Robot Learning*, volume 155 of *Proceedings of Machine Learning Research*, pp. 264–285. PMLR, 2021.

A TRAINING DYNAMICS

Figure 4 reports per-epoch ADE and FDE for TraffusionGen and the three baselines. TraffusionGen reaches the lower-error regime earlier and remains below the baselines for most epochs. Both curves show the improvement over the course of optimization, complementing the final comparisons in the main text.

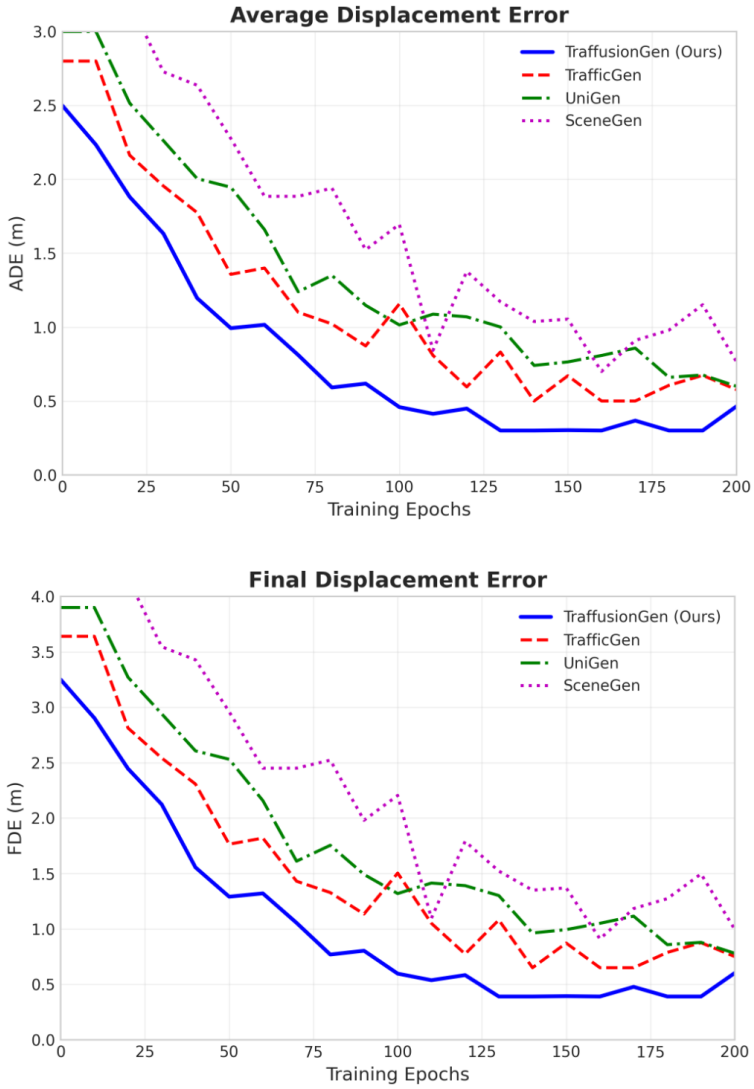


Figure 4: **Optimization behavior.** Per-epoch ADE (top) and FDE (bottom). TraffusionGen enters the low-error regime earlier and remains below the baselines for most epochs.