World4Drive: End-to-End Autonomous Driving via Intention-aware Physical Latent World Model

Supplementary Material

1. Detailed Implementation

1.1. nuScenes Benchmark

1.1.1. Detailed Metrics

Following prior methods, nuScenes evaluates trajectories using **L2 Error** and **Collision Rate**. The **L2 Error** measures the distance between the planned trajectory and the human-driven trajectory. The **Collision Rate** measures how frequently the planned trajectory collides with other agents on the road.

1.1.2. Data Processing Pipeline

In nuScenes experiments, the original 6-surround-view images have a resolution of 900×1600 . They are first resized to 360×640, then padded to a final input size of 384×640, yielding $I_t^{img} \in \mathbb{R}^{6\times3\times384\times640}$. After processing through the image backbone network, the image features are downsampled to a spatial resolution of 12×20, producing a feature $I_t^f \in \mathbb{R}^{6\times240\times256}$, which serves as the input for subsequent network layers.

1.2. NAVSIM Benchmark

1.2.1. Detailed Metrics

NAVSIM evaluates trajectories using PDMS, with the specific evaluation formula given as:

$$\text{PDMS} = \text{NC} \times \text{DAC} \times \frac{(5 \times \text{EP} + 5 \times \text{TTC} + 2 \times \text{Comf.})}{12} \tag{1}$$

NC and DAC serve as multiplicative factors for other evaluation metrics. NC assesses the likelihood of the ego vehicle colliding with other agents, while DAC evaluates whether the predicted future trajectory remains within the drivable region. Other metrics are aggregated through a weighted summation. EP quantifies the ego vehicle's anticipated driving distance along the designated route within the next 4 seconds. TTC measures the safety of the current state by estimating the time required for a potential collision with other agents if the vehicle maintains its present motion state. Comf. assesses driving smoothness by analyzing acceleration, heading changes, and other dynamic factors.

1.2.2. Data Processing Pipeline

In the NAVSIM experiments, the input features consist of ego status e_t and image information $I_t^{final} \in \mathbb{R}^{3\times256\times1024}$. e_t is a vector of $[c_t, v_t, a_t]$, where v_t and a_t denote the ego vehicle's velocity and acceleration at time t, and c_t is a one-hot encoded vector indicating the control command (left turn, right turn, or straight). The image input, originally of size $I_t^{\text{raw}} \in \mathbb{R}^{3\times1080\times1920}$, undergoes a structured preprocessing pipeline. The front view is vertically cropped, resulting in $I_t^f \in \mathbb{R}^{3\times1024\times1920}$, while the left and right views are cropped along both height and width dimensions, yielding $I_t^l, I_t^r \in \mathbb{R}^{3\times1024\times1088}$. The processed views are then concatenated along the width dimension, forming

$$I_t^{\text{concat}} = \text{Concat}(I_t^l, I_t^f, I_t^r) \tag{2}$$

Finally, the concatenated image is resized to

$$I_t^{\text{final}} = \text{Resize}(I_t^{\text{concat}}, (3, 256, 1024)) \tag{3}$$

Table 1. Ablation study of timestamp interval

Timestamp interval n	L2 (m) ↓	Collsion (%) ↓
1	0.55	0.35
3	**0.50**	**0.16**
6	0.52	0.24

2. Additional Experimental Results

2.1. Ablation of Timestamp Interval

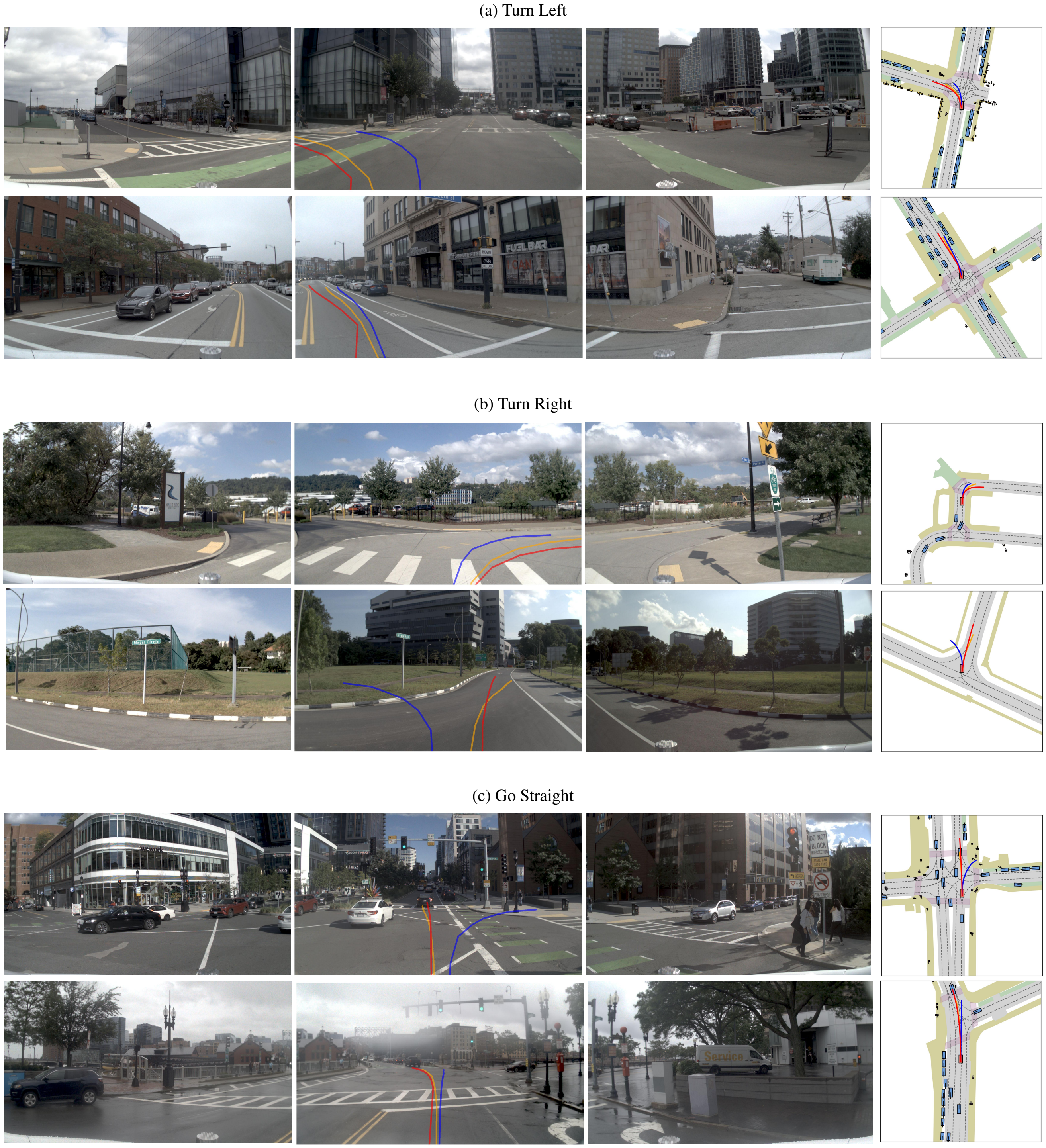
In this section, we investigate the impact of varying the timestamp interval n of the input data on model performance. Table 1 shows the quantitative results of our ablation experiments.

As shown in Table 1, the model achieves optimal performance when the timestamp interval is set to 3, obtaining the lowest L2 error of 0.50 meters and the lowest collision rate of 0.16%. Increasing the interval to 6 results in a slight performance degradation, with the L2 error rising to 0.52 meters and the collision rate increasing to 0.24%. Conversely, using a very short interval ($n = 1$) also negatively affects performance, causing an increase in both L2 error (0.55 meters) and collision rate (0.35%).

Experimental results show that the best performance is achieved when $n = 3$. We argue that when n is too small, the scene information lacks significant variation, preventing the model from capturing sufficient contextual temporal information. On the other hand, when the timestamp span is too large, the model's assessment of future scenes becomes less accurate.

2.2. Ablation of Reconstruction Losses

Table 2 presents an ablation study evaluating the influence of different reconstruction losses, specifically Mean



(a) Turn Left

(b) Turn Right

(c) Go Straight

Figure 1. Red trajectory is ground-truth, while orange and blue trajectories are generated by World4Drive and LAW [**?**].

Squared Error (MSE), Kullback-Leibler (KL) divergence, and Cosine similarity loss. In general, the choice of reconstruction loss has a relatively minor impact on the model's overall performance. Thus, We choose the MSE loss as the reconstruction loss.

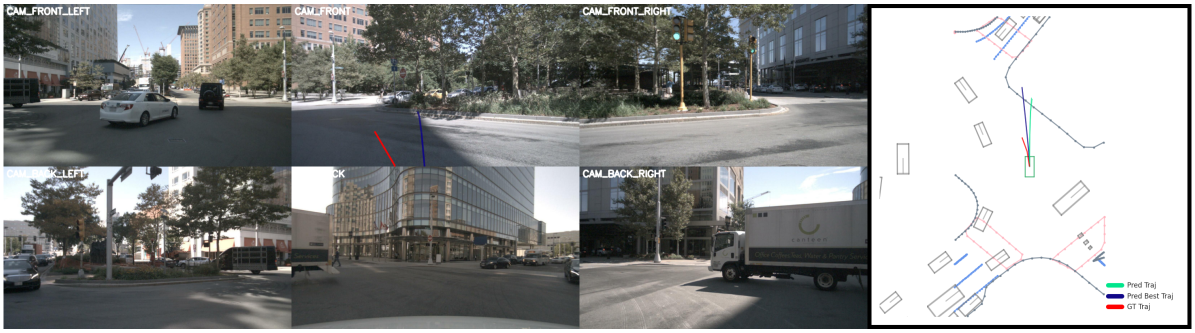


Figure 2. Red trajectory is ground-truth. Green trajectories and blue trajectory are generated by World4Drive, while blue one is the best one selected by world model.

Table 2. Ablation study of reconstruction losses

ID	Mse	KL	Cosine	L2(m)↓	Collsion(%) ↓
1	✓			0.50	0.16
2		✓		0.50	0.17
3			✓	0.52	0.20
4	✓	✓		0.51	0.18
5	✓		✓	0.51	0.19
6	✓	✓	✓	**0.50**	**0.16**

Table 3. Ablation study of Metric Depth Models

Metric Depth Model	L2 (m) ↓	Collsion (%) ↓
Metirc3D v1 Convtiny	0.50	0.19
Metirc3D v2 Small	0.51	0.20
Metirc3D v2 Giant	**0.50**	**0.16**

2.3. Ablation of Metric Depth Models

In this section, we explore how different metric depth models affect the model's performance. Table 3 summarizes our findings. We observe that all tested depth models (Convtiny, Small, Giant) achieve comparable performance. Specifically, both the Convtiny and Giant models achieve the lowest L2 error of 0.50 meters, while the giant model slightly outperforms in collision rate (0.16%) compared to Convtiny (0.19%) and small (0.20%). However, the performance differences among these depth models are minimal. This indicates that while the inclusion of metric depth information significantly improves overall model performance, the model's complexity or size is not the primary determinant of accuracy or collision avoidance capabilities. Thus, even simpler depth models can provide sufficient spatial context to achieve similar trajectory prediction results. Given that the performance of different depth models is comparable, in practical deployment, an appropriate depth model can be selected based on specific computational requirements.

3. Additional Qulitative Results

3.1. More Visulization

We provide high-quality visualizations on NASIM benchmarks, as shown in Figure 1. We present a comparison between LAW and World4Drive in scenarios involving left turns, right turns, and straight driving. Compared to LAW, World4Drive demonstrates a stronger alignment with the ground-truth trajectory. Additionally, World4Drive is better able to capture scene information, particularly in determining whether the driving trajectory lies within the drivable area.

3.2. Failure Case

We present failure cases of World4Drive on NuScenes in Figure 2, showing low-quality trajectories under incorrect driving commands. We found misannotations in NuScenes, such as a left-turn scene labeled as go-straight in Figure 2, making it difficult to generate accurate trajectory.