CONCEPTWORLDMODEL: CAN WORLD MODELS LEARN DYNAMICS IN CONCEPT SPACE?

Anonymous authors
Paper under double-blind review

ABSTRACT

World models have demonstrated substantial capability in predicting future states in complex environments; however, their internal representations often remain difficult to interpret. Consequently, it remains unclear whether such models merely encode information required for prediction or develop structured semantic representations of the surrounding environment. This work investigates whether higher-order semantic information can become recoverable from hierarchical representations built from simpler grounded concepts without direct supervision of those higher-order concepts. To address this question, ConceptWorldModel introduces a hierarchical world model for autonomous driving that transforms visual observations into grounded concepts, relational representations, and higher-order scene representations. These representations define a structured concept space that is propagated through action-conditioned temporal dynamics, enabling recursive prediction of future concepts across multiple horizons. The model is evaluated on Bench2Drive using a structured concept ontology derived from driving-scene annotations and metadata. Experimental results demonstrate sustained predictive performance over extended rollouts, achieving an average precision of 0.783 for binary concepts and a normalized mean absolute error of 0.435 across scalar regression targets at a 20-step prediction horizon. Evaluation of concepts excluded from direct supervision further shows that higher-order semantic information remains recoverable across the learned representation hierarchy. Overall, the results show that future driving states can be represented and predicted through an explicit semantic concept space, providing a structured alternative to exclusively opaque latent world-model representations.

1 INTRODUCTION

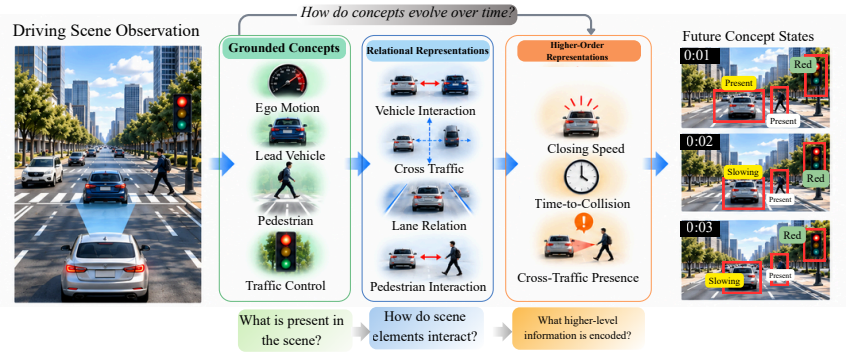


Figure 1: High-level overview of ConceptWorldModel. Visual observations are organized into grounded, relational, and higher-order semantic representations that define a structured concept space and are evolved through action-conditioned dynamics for future concept prediction.

World models aim to provide intelligent agents with an internal predictive representation of their environment, capturing both its current state and how that state may evolve over time (Ding et al.,

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.

2025). By learning environment dynamics, these models allow agents to anticipate future states and evaluate the consequences of actions before execution (M. Moerland et al., 2023). This capability has established world modeling as an important paradigm for sequential decision making in robotics, embodied intelligence, simulation, and autonomous systems, where prediction is closely coupled with planning and control (Ding et al., 2025; Kirchner et al., 2026).

World-model research has developed along several complementary directions. Latent-dynamics and imagination-based approaches learn compact representations in which future trajectories can be predicted and used for behavior learning or control. PlaNet performs planning directly in a learned latent state space (Hafner et al., 2019b), while Dreamer and its extensions learn policies from imagined trajectories (Hafner et al., 2019a; 2025). Task-oriented approaches such as MuZero, TD-MPC, and TD-MPC2 learn predictive representations tailored to search or continuous control (Schrittwieser et al., 2020; Hansen et al., 2022; 2024). Generative approaches instead model future observations, including autoregressive visual tokens in IRIS (Micheli et al., 2022), diffusion-based prediction in DIAMOND (Alonso et al., 2024), and action-controllable environments in Genie (Bruce et al., 2024). In autonomous driving, MILE, DriveWorld, OccWorld, and World4Drive model future scene evolution for prediction and planning (Hu et al., 2022; Min et al., 2024; Zheng et al., 2024; 2025). Collectively, these approaches demonstrate the effectiveness of world models, but the semantic organization of their learned states is generally left implicit.

Despite these advances, understanding what information is encoded within a world model remains challenging. Many approaches rely on latent representations shaped primarily by predictive, reconstruction, or control objectives rather than explicit semantic organization (Ding et al., 2025). Successful prediction therefore does not reveal how information about scene elements, their interactions, and future events is organized within the predictive state. It consequently remains unclear whether world-model dynamics can operate over semantically structured representations while preserving information associated with higher-order scene properties.

This question is particularly relevant to autonomous driving, where effective prediction depends not only on individual scene elements but also on their relationships. Information such as whether a lead vehicle is closing in, whether cross traffic is present, or how collision risk evolves depends on multiple lower-level observations and their temporal interactions. Driving therefore provides a natural setting for studying world dynamics in a structured concept space. However, existing driving world models primarily emphasize future scene prediction, occupancy forecasting, latent dynamics, or trajectory planning, leaving the organization and temporal evolution of semantic information comparatively unexplored.

The aim of this study is therefore to investigate whether world-model dynamics can be learned over a structured concept space while preserving higher-order semantic information that is not directly supervised. To address this question, ConceptWorldModel introduces an explicit hierarchy of grounded, relational, and higher-order representations and models their temporal evolution under driving actions, as illustrated in Fig. 1. The main contributions of this work are summarized as follows:

- A concept-level annotation schema derived from Bench2Drive that represents semantically meaningful driving-scene properties and supports the study of grounded, relational, and higher-order semantic information.
- ConceptWorldModel, a hierarchical semantic world model that organizes scene information into a structured concept space and evolves the resulting state through action-conditioned temporal dynamics.
- A representation-level evaluation framework that examines how semantic information is distributed across the hierarchy using concepts excluded from direct supervision and measuring their recoverability from different representation levels.
- An empirical evaluation combining multi-step future concept prediction, held-out concept probing, controlled comparisons with alternative world-model dynamics, architectural ablations, and counterfactual action analysis across increasing prediction horizons.

The remainder of this paper is organized as follows. Section 2 reviews related work, Section 3 presents the proposed method, Section 4 presents the experimental results, and Section 5 concludes the paper.

2 Related Work

2.1 World Models and Sequential Prediction

Latent-dynamics methods such as PlaNet, Dreamer, and DreamerV3 learn compact predictive representations for imagination and control (Hafner et al., 2019b;a; 2025), while MuZero, TD-MPC, and TD-MPC2 optimize learned dynamics for search and trajectory planning (Schrittwieser et al., 2020; Hansen et al., 2022; 2024). Generative alternatives instead model future observations: IRIS predicts discrete visual tokens, whereas DIAMOND and Genie use diffusion-based or action-controllable generation (Micheli et al., 2022; Alonso et al., 2024; Bruce et al., 2024). Despite strong predictive and control performance, these approaches generally represent future dynamics in latent or observation spaces whose semantic organization remains largely implicit.

2.2 World Models for Autonomous Driving

Driving-specific world models likewise emphasize prediction and planning. MILE jointly learns latent dynamics and a driving policy (Hu et al., 2022), DriveWorld models spatiotemporal scene representations (Min et al., 2024), and OccWorld predicts future 3D occupancy (Zheng et al., 2024). Drive-WM and Vista focus on controllable future generation (Wang et al., 2024; Gao et al., 2024), while World4Drive and GenieDrive incorporate intention-aware planning and occupancy-guided physical structure (Zheng et al., 2025; Yang et al., 2026). These methods substantially enrich scene and temporal modeling, but they primarily evolve latent, occupancy, or visual representations rather than an explicitly structured concept space whose semantic components can be directly examined.

2.3 Concept-Based and Structured Representation Learning

Concept Bottleneck Models expose human-defined intermediate concepts for inspection and intervention (Koh et al., 2020), while Concept Embedding Models represent concepts through richer embeddings (Espinosa Zarlenga et al., 2022). Post-hoc and label-free approaches reduce reliance on dense concept annotations (Yuksekgonul et al., 2022; Oikarinen et al., 2023), and neural-symbolic methods model relations among semantic concepts (Barbiero et al., 2023). Related structured approaches include Neural Relational Inference, which learns latent interaction graphs (Kipf et al., 2018), Slot Attention, which discovers object-centric representations (Locatello et al., 2020), and SlotFormer, which extends slot representations to temporal prediction (Wu et al., 2022).

These works establish the value of interpretable, factorized, and relational representations, but typically do not study action-conditioned world dynamics over an explicit hierarchy of grounded, relational, and higher-order semantic representations. ConceptWorldModel instead constructs a structured concept space from grounded driving factors, models its temporal evolution under actions, and examines whether semantic information withheld from direct supervision remains recoverable across its representation levels.

3 Methodology

ConceptWorldModel (CWM) learns a structured semantic state for dynamic driving scenes and models its evolution through time. As illustrated in Figure 2, the architecture comprises four stages: visual representation, concept grounding, hierarchical concept abstraction, and temporal concept dynamics. A frozen visual encoder extracts spatial features from a short history of camera observations, the Concept Grounding Module (CGM) produces factor-specific grounded representations, and the Hierarchical Concept Abstraction Module (HCAM) organizes them into relational and higher-order representations. Together, these levels define a structured concept space. The Temporal Concept Dynamics Module (TCDM) then evolves this state conditioned on ego actions and navigation context, enabling recursive prediction of future semantic states.

3.1 Dataset

The experiments use the training set of **Bench2Drive-V0.0.4**, an autonomous-driving benchmark developed in the CARLA simulator for evaluating interactive driving scenarios Jia et al. (2024). The

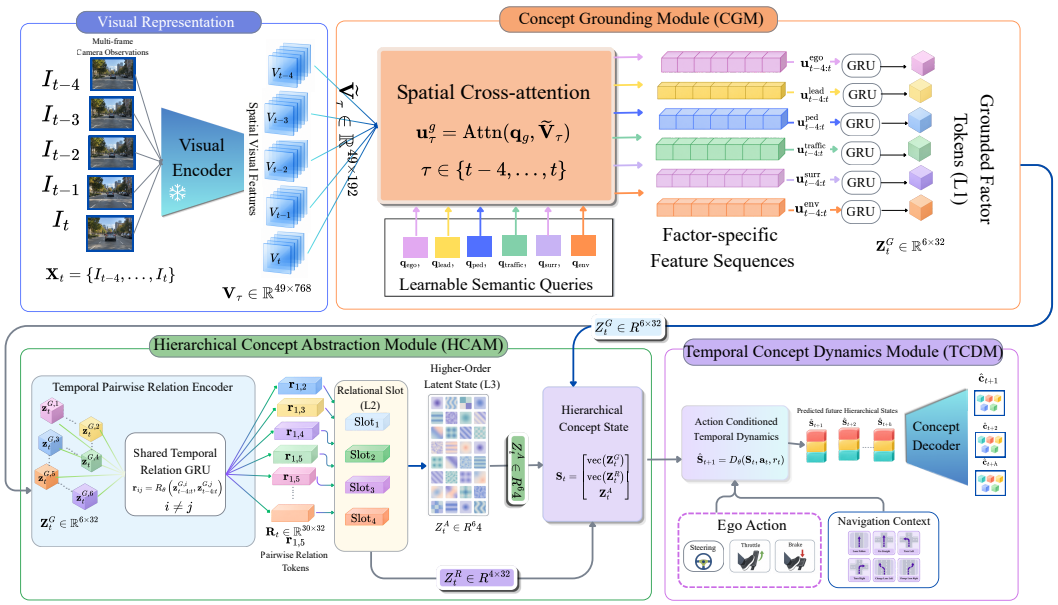


Figure 2: Overview of ConceptWorldModel (CWM). Visual observations are transformed into grounded, relational, and higher-order semantic representations that define a structured concept space. Action- and route-conditioned temporal dynamics recursively evolve this state to predict future concepts.

training set contains 44 scenarios with 25 routes each, yielding 1,100 routes in total.[1] Bench2Drive provides temporally ordered camera observations, frame-level annotations, and driving measurements. CWM uses synchronized visual observations together with ego-vehicle control signals comprising steering, throttle, and brake. Route commands are retained separately as exogenous navigation context rather than treated as visually grounded scene concepts.

3.2 Concept Extraction and Ontology

The concept extraction pipeline transforms Bench2Drive frame-level annotations and route metadata Jia et al. (2024) into a structured semantic description of each driving scene. The ontology comprises 16 semantic concepts and one contextual variable spanning ego motion, lead-vehicle and pedestrian interactions, surrounding traffic, traffic-control state, environmental conditions, and navigation context. Across the 17 variables, seven are continuous, one is a count variable, six are binary, and three are categorical. Route Command is retained separately as exogenous navigation context.

The ontology combines directly annotated quantities with spatially and temporally derived concepts. Spatial concepts are constructed from the ego-relative positions, orientations, and states of surrounding actors, whereas temporal concepts use consecutive observations to characterize changes in motion and interaction. Ego acceleration is derived from successive speed measurements, while lead-vehicle closing speed and time-to-collision are derived from changes in relative lead distance. Actor identity is preserved across time when the required identifiers are available, and quantities that are undefined for a scene are treated as unavailable rather than assigned artificial semantic values. The complete ontology, including variable definitions, derivations, and types, is provided in Appendix A (Table 4).

3.3 Visual Representation

CWM encodes a five-frame history of camera observations while preserving the spatial structure required for concept grounding. Rather than reducing each frame to a single global embedding,

[1] https://github.com/Thinklab-SJTU/Bench2Drive/tree/0.0.4

the visual front-end retains spatial features that allow different semantic factors to attend to distinct image regions.

A pretrained SigLIP2-B/16 visual encoder Tschannen et al. (2025) processes each frame independently and remains frozen throughout training:

$$\mathbf{X}_t = \{I_{t-4}, \ldots, I_t\}, \tag{1}$$

$$\mathbf{V}_{t-4:t} = \{E_{\text{vis}}(I_\tau)\}_{\tau=t-4}^{t}. \tag{2}$$

Here, $\mathbf{X}_t$ denotes the five-frame observation history, E_{vis} denotes the frozen visual encoder, and $\mathbf{V}_{t-4:t}$ denotes the temporally ordered sequence of spatial feature representations provided to the Concept Grounding Module.

3.4 Concept Grounding Module

The **Concept Grounding Module (CGM)** transforms spatial visual features into grounded representations of six semantic factors: *ego motion*, *lead vehicle*, *pedestrian*, *traffic control*, *surrounding traffic*, and *environment*. Grouping related concepts within shared factors provides a compact semantic basis for the structured concept space.

For each frame, visual features are projected into a shared latent space and augmented with spatial positional encoding. A learnable query for each factor attends to the projected feature map to retrieve factor-specific visual evidence, which is aggregated across the five-frame history by a factor-specific GRU:

$$\begin{aligned}
\widetilde{\mathbf{V}}_\tau &= P(\mathbf{V}_\tau) + \mathbf{P}_{\text{pos}}, \\
\mathbf{u}_\tau^g &= \text{Attn}\left(\mathbf{q}_g, \widetilde{\mathbf{V}}_\tau\right), \qquad \tau \in \{t-4, \ldots, t\}, \\
\mathbf{z}_\tau^{G,g} &= \text{GRU}_g\left(\mathbf{u}_{t-4:\tau}^g\right), \qquad g = 1, \ldots, 6, \\
\mathbf{Z}_t^G &= \left[\mathbf{z}_t^{G,1}, \ldots, \mathbf{z}_t^{G,6}\right].
\end{aligned} \tag{3}$$

Here, $P(\cdot)$ denotes the learned feature projection, $\mathbf{P}_{\text{pos}}$ the spatial positional encoding, $\mathbf{q}_g$ the learnable query for factor g, and $\mathbf{z}_t^{G,g}$ its temporally aggregated grounded representation. The sequence $\mathbf{z}_{t-4:t}^{G,g}$ denotes the corresponding GRU hidden states across the observation history.

Of the 17 ontology variables, 13 atomic concepts directly supervise CGM. *Lead-Vehicle Closing Speed*, *Cross-Traffic Presence*, and *Time-to-Collision* are withheld from concept-level supervision and used exclusively as held-out semantic probes, while *Route Command* is retained as exogenous navigation context. The complete assignment of concepts to semantic factors is provided in Appendix B (Table 5).

For each supervised atomic concept c, a concept-specific prediction head operates on its associated grounded factor:

$$\hat{y}_t^c = \phi_c\left(\mathbf{z}_t^{G,g(c)}\right), \qquad c \in \mathcal{C}_{\text{atom}}, \tag{4}$$

where $g(c)$ maps concept c to its semantic factor and ϕ_c denotes its prediction head. These heads provide direct semantic supervision, while the grounded factor representations are passed to the subsequent hierarchical stages.

3.5 Hierarchical Concept Abstraction Module

The **Hierarchical Concept Abstraction Module (HCAM)** organizes grounded semantic factors into relational and higher-order representations. Many driving properties depend on interactions among multiple scene factors rather than on any factor in isolation. HCAM therefore first models pairwise temporal relationships between the six grounded factor trajectories and subsequently aggregates these interactions into a compact higher-order representation.

For each ordered pair of distinct factors (i, j), a shared temporal relation encoder produces

$$\begin{aligned} \mathbf{r}_{ij} &= R_\theta\left(\mathbf{z}^{G,i}_{t-4:t}, \mathbf{z}^{G,j}_{t-4:t}\right), \qquad i \neq j, \\ \mathbf{R}_t &= \left[\mathbf{r}_{ij}\right]_{i\neq j}, \\ \mathbf{Z}^R_t &= \mathrm{SlotAttn}\left(\mathbf{R}_t\right), \\ \mathbf{Z}^A_t &= F_{\mathrm{abs}}\left(\mathrm{vec}\left(\mathbf{Z}^R_t\right)\right). \end{aligned} \tag{5}$$

With six grounded factors and self-relations excluded, $\mathbf{R}_t$ contains 30 ordered pairwise relation tokens. Competitive slot attention compresses these relations into four learned relational slots $\mathbf{Z}^R_t$. The slots receive neither predefined semantic labels nor direct concept supervision, allowing them to aggregate recurring interaction patterns from the relation tokens. Finally, F_{abs} maps the relational slots to the higher-order latent representation $\mathbf{Z}^A_t$.

Together, these representations define three levels of the structured concept space: grounded factors $\mathbf{Z}^G_t$, relational representations $\mathbf{Z}^R_t$ derived from pairwise interactions $\mathbf{R}_t$, and the higher-order representation $\mathbf{Z}^A_t$.

3.6 Temporal Concept Dynamics Module

The **Temporal Concept Dynamics Module (TCDM)** models the future evolution of the structured concept space. Rather than predicting individual concepts independently, TCDM evolves a joint state containing grounded, relational, and higher-order information while conditioning each transition on ego control and navigation context.

At time t, the three representation levels form the structured concept state

$$\mathbf{S}_t = \left[\mathrm{vec}(\mathbf{Z}^G_t); \mathrm{vec}(\mathbf{Z}^R_t); \mathbf{Z}^A_t\right]. \tag{6}$$

The resulting state is 384-D, comprising six 32-D grounded tokens (192 dimensions), four 32-D relational slots (128 dimensions), and a 64-D higher-order representation. Future states are generated recursively as

$$\hat{\mathbf{S}}_{t+h} = D_\theta\left(\hat{\mathbf{S}}_{t+h-1}, \mathbf{a}_{t+h-1}, \mathbf{c}_{t+h-1}\right), \qquad h \geq 1, \qquad \hat{\mathbf{S}}_t = \mathbf{S}_t, \tag{7}$$

where $\mathbf{a}_t$ denotes ego-vehicle control, $\mathbf{c}_t$ denotes navigation context, and D_θ denotes the learned temporal dynamics model. A future concept decoder maps each recursively predicted state to the supervised semantic concept space:

$$\hat{\mathbf{y}}_{t+h} = G_\psi\left(\hat{\mathbf{S}}_{t+h}\right), \tag{8}$$

where G_ψ denotes the future concept decoder.

3.7 Training and Baselines

Following stage-wise initialization, all trainable CWM modules are jointly optimized while the visual encoder remains frozen. The training objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{ground}} + \mathcal{L}_{\mathrm{roll}} + 0.2\mathcal{L}_{\mathrm{direct}} + 0.01\mathcal{L}_{\mathrm{slot}} + 0.05\mathcal{L}_{\mathrm{peq}}, \tag{9}$$

where $\mathcal{L}_{\mathrm{ground}}$ supervises the 13 atomic concepts, $\mathcal{L}_{\mathrm{roll}}$ supervises recursive future concept prediction, $\mathcal{L}_{\mathrm{direct}}$ supervises direct future prediction from the hierarchical representation, and $\mathcal{L}_{\mathrm{slot}}$ encourages diversity among the relational slots. The predictive-equivalence term $\mathcal{L}_{\mathrm{peq}}$ encourages the pairwise geometry of L_3 to reflect similarity in future atomic behavior for samples undergoing similar action sequences. Undefined concept targets are masked from their corresponding loss terms.

Optimization uses AdamW with validation-based checkpoint selection and early stopping. Recursive prediction is introduced through a progressive rollout curriculum of 5, 10, and 20 steps. The 1,100 routes are partitioned into 880/110/110 disjoint training, validation, and test routes before temporal windows are constructed. Reported dynamics results are averaged across three random seeds.

For controlled comparison, DreamerV3 Hafner et al. (2025), TD-MPC2 Hansen et al. (2024), and MILE Hu et al. (2022) are adapted as latent-dynamics baselines. Each baseline operates on the same fixed 384-D CWM state and uses identical ego actions, navigation context, data partitions, rollout horizons, and semantic evaluation protocol. This isolates differences in temporal dynamics while holding the input state and downstream concept evaluation fixed.

3.8 Evaluation Protocol

CWM is evaluated over recursive prediction horizons $h \in \{1, 5, 10, 20\}$. Continuous targets are standardized using training-set statistics, $\tilde{y}_c = (y_c - \mu_c)/\sigma_c$, and nMAE is computed as MAE in this standardized space and macro-averaged with equal weight across the six supervised scalar regression targets. Binary concepts use Average Precision (AP), and categorical concepts use accuracy.

Semantic organization is evaluated using the three concepts withheld from direct supervision: *Lead-Vehicle Closing Speed*, *Cross-Traffic Presence*, and *Time-to-Collision*. Recoverability is measured independently from L_1, L_2, and L_3. To control for dimensionality, L_1 and L_2 are standardized and reduced to 64 dimensions using training-fitted PCA, while the native 64-D L_3 is retained. Continuous probes use Ridge regression with nMAE, and the binary probe uses class-balanced logistic regression with AP.

4 Results

4.1 Long-Horizon Concept Prediction

A central question is whether CWM can preserve semantically meaningful scene information as predictions are recursively propagated into the future. This is evaluated across increasing horizons and compared with TD-MPC2, MILE, and DreamerV3 under the same state representation, control inputs, route context, data partitions, and concept readout. As shown in Table 1, CWM maintains stronger concept prediction performance across all evaluated horizons.

Table 1: Controlled latent-dynamics comparison across recursive prediction horizons. Lower nMAE and higher AP and categorical accuracy indicate better performance. DreamerV3 continuous prediction exhibited numerical divergence under the controlled adaptation and is therefore reported as "div.".

	$H = 1$			$H = 5$			$H = 10$			$H = 20$		
Model	**nMAE ↓**	**AP ↑**	**Acc. ↑**	**nMAE ↓**	**AP ↑**	**Acc. ↑**	**nMAE ↓**	**AP ↑**	**Acc. ↑**	**nMAE ↓**	**AP ↑**	**Acc. ↑**
CWM	**0.410**	**0.852**	**0.745**	**0.376**	**0.849**	**0.750**	**0.390**	**0.831**	**0.748**	**0.435**	**0.783**	**0.737**
TD-MPC2	0.876	0.745	0.297	0.942	0.730	0.229	0.972	0.698	0.222	1.020	0.635	0.226
MILE	6.715	0.374	0.091	6.475	0.370	0.089	6.463	0.367	0.082	6.549	0.357	0.077
DreamerV3	div.	0.441	0.374	div.	0.421	0.331	div.	0.412	0.354	div.	0.389	0.369

4.2 Hierarchical Semantic Specialization

The learned hierarchy is examined to determine how semantic information associated with concepts withheld from direct supervision is distributed across successive representation levels. Recoverability is assessed independently from the grounded (L_1), relational (L_2), and higher-order (L_3) representations under the capacity-controlled probing protocol described in Sec. 3.8. The resulting probe performance is summarized in Table 2.

The held-out concepts exhibit different recoverability patterns across the hierarchy. Lead-vehicle closing speed and cross-traffic presence are most recoverable from the relational representation L_2, while time-to-collision attains its lowest error at the higher-order representation L_3. The absence of a uniform monotonic trend shows that increasing representation depth does not make every held-out concept progressively more accessible. Instead, different semantic quantities are most recoverable at different levels of the learned hierarchy. A qualitative visualization of this layer-dependent organization is provided in Fig. 3.

Table 2: Capacity-controlled recovery of held-out concepts across the CWM hierarchy. Lower nMAE and higher AP indicate better performance.

Held-out concept	**Metric**	L_1	L_2	L_3
Lead-Vehicle Closing Speed	nMAE ↓	0.539	**0.509**	0.512
Cross-Traffic Presence	AP ↑	0.843	**0.872**	0.861
Time-to-Collision	nMAE ↓	0.587	0.569	**0.555**

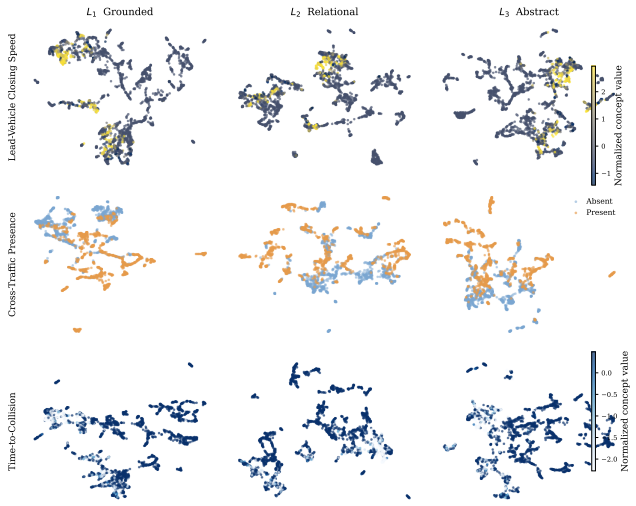


Figure 3: UMAP visualization of held-out concept organization across the CWM hierarchy. Each column shows a two-dimensional UMAP projection of the same test observations represented at the grounded (L_1), relational (L_2), and higher-order (L_3) levels. Points are colored according to the corresponding held-out concept value or class.

4.3 Architecture Ablation Analysis

The architecture ablations assess the contribution of the principal CWM components to recursive concept prediction across increasing rollout horizons. Table 3 compares the complete model with variants that remove action conditioning, replace the hierarchy with a flat latent state, retain only the grounded representation, remove temporal relation modeling, or omit the predictive-equivalence objective.

Table 3: Architecture ablations across recursive prediction horizons. Lower nMAE and higher AP and accuracy indicate better performance.

	$H = 1$			$H = 5$			$H = 10$			$H = 20$		
Variant	**nMAE ↓**	**AP ↑**	**Acc. ↑**	**nMAE ↓**	**AP ↑**	**Acc. ↑**	**nMAE ↓**	**AP ↑**	**Acc. ↑**	**nMAE ↓**	**AP ↑**	**Acc. ↑**
Full CWM	0.410	**0.852**	**0.745**	0.376	**0.849**	**0.750**	0.390	**0.831**	**0.748**	0.435	0.783	**0.737**
No action conditioning	0.414	0.844	0.717	0.411	0.835	0.712	0.425	0.815	0.696	0.468	0.775	0.663
Flat 384-D	**0.407**	0.847	0.737	0.379	0.843	0.736	0.413	0.824	0.724	0.473	0.778	0.699
L_1 only	0.412	0.851	0.726	0.388	0.842	0.742	0.422	0.823	0.746	0.476	0.772	0.730
Static relations	0.412	0.844	0.722	0.374	0.842	0.729	0.378	0.828	0.729	0.421	**0.794**	0.717
No predictive-equivalence	0.412	0.842	0.728	**0.372**	0.840	0.731	**0.373**	0.826	0.728	**0.412**	0.792	0.710

The complete architecture provides the most consistent binary and categorical performance, achieving the highest AP at $H = 1$, $H = 5$, and $H = 10$ and the highest categorical accuracy at all horizons. Removing action conditioning produces increasing degradation with rollout depth; at $H = 20$, nMAE rises from 0.435 to 0.468 and categorical accuracy decreases from 0.737 to 0.663, indicating the growing importance of control information during recursive prediction.

The hierarchy contributes most clearly at longer horizons. Although the flat 384-D variant attains slightly lower nMAE at $H = 1$, its error reaches 0.473 at $H = 20$, compared with 0.435 for CWM, while its AP and categorical accuracy are lower. Similarly, the L_1-only variant reaches an nMAE of 0.476 at $H = 20$. These results indicate that relational and higher-order representations support long-horizon semantic prediction beyond grounded features or matched latent dimensionality alone.

The remaining ablations reveal metric-dependent trade-offs. Removing predictive equivalence yields lower nMAE from $H = 5$ onward, while static relations attain the highest AP at $H = 20$; however, both variants reduce categorical accuracy relative to the full model. Thus, temporal relation modeling and predictive equivalence do not uniformly improve every target type. Overall, the clearest long-horizon gains arise from action conditioning and the hierarchical representation.

4.4 Counterfactual Action Sensitivity

Counterfactual action sensitivity is evaluated by holding the observed scene fixed while replacing the future control sequence with fixed braking $(0, 0, 1)$, coasting $(0, 0, 0)$, or throttle $(0, 1, 0)$ commands, reported as $(\text{steer}, \text{throttle}, \text{brake})$. Each command is standardized using training-set action statistics and held constant throughout the rollout. Figure 4 shows the resulting ego-speed predictions and the fraction of test examples satisfying the expected ordering across rollout horizons.

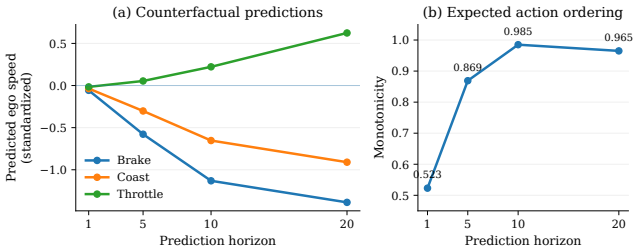


Figure 4: Counterfactual action sensitivity across recursive prediction horizons. Left: predicted ego speed under braking, coasting, and throttle interventions in standardized target space. Right: fraction of test examples satisfying $\hat{v}_{\text{brake}} \leq \hat{v}_{\text{coast}} \leq \hat{v}_{\text{throttle}}$.

The intervention trajectories become increasingly differentiated with rollout depth. At $H = 20$, standardized ego-speed predictions are -1.387, -0.910, and 0.624 under braking, coasting, and throttle, respectively. Consistency with the expected action ordering increases from 0.523 at $H = 1$ to 0.869 at $H = 5$ and 0.985 at $H = 10$, remaining high at 0.965 at $H = 20$.

These results complement the action-conditioning ablation by showing that control inputs affect the direction of predicted state evolution rather than only aggregate prediction performance. The learned dynamics therefore respond systematically to alternative control sequences over recursive rollouts.

5 Conclusion

ConceptWorldModel introduces a semantic world-model formulation that organizes grounded driving factors into relational and higher-order representations and models their evolution through action-conditioned temporal dynamics. The evaluation combines recursive concept prediction, held-out semantic probing, controlled baseline comparisons, architectural ablations, and counterfactual action interventions. The results show that the hierarchical representation supports long-horizon semantic prediction and that information excluded from direct supervision remains recoverable across representation levels. Action conditioning becomes increasingly important with rollout depth, while counterfactual interventions show that the learned dynamics respond systematically to alternative control sequences. The ablations further indicate that individual architectural components contribute differently across continuous, binary, and categorical targets. Overall, ConceptWorldModel provides a structured framework for learning and evolving dynamics in concept space, connecting grounded perception, relational representations, higher-order semantic information, and action-conditioned future prediction. This enables world models to be examined not only through predictive performance, but also through how semantic information is organized and propagated over time.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript editing and organization.

REFERENCES

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. *Advances in Neural Information Processing Systems*, 37:58757–58791, 2024.

Pietro Barbiero, Gabriele Ciravegna, Francesco Giannini, Mateo Espinosa Zarlenga, Lucie Charlotte Magister, Alberto Tonda, Pietro Lió, Frederic Precioso, Mateja Jamnik, and Giuseppe Marra. Interpretable neural-symbolic concept reasoning. In *International Conference on Machine Learning*, pp. 1801–1825. PMLR, 2023.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first international conference on machine learning*, 2024.

Jingtao Ding, Yunke Zhang, Yu Shang, Yuheng Zhang, Zefang Zong, Jie Feng, Yuan Yuan, Hongyuan Su, Nian Li, Nicholas Sukiennik, et al. Understanding world or predicting future? a comprehensive survey of world models. *ACM Computing Surveys*, 58(3):1–38, 2025.

Mateo Espinosa Zarlenga, Pietro Barbiero, Gabriele Ciravegna, Giuseppe Marra, Francesco Giannini, Michelangelo Diligenti, Zohreh Shams, Frederic Precioso, Stefano Melacci, Adrian Weller, et al. Concept embedding models: Beyond the accuracy-explainability trade-off. *Advances in neural information processing systems*, 35:21400–21413, 2022.

Shenyuan Gao, Jiazhi Yang, Li Chen, Kashyap Chitta, Yihang Qiu, Andreas Geiger, Jun Zhang, and Hongyang Li. Vista: A generalizable driving world model with high fidelity and versatile controllability. *Advances in Neural Information Processing Systems*, 37:91560–91596, 2024.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019a.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019b.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025.

Nick Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, volume 2024, pp. 47376–47405, 2024.

Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. *arXiv preprint arXiv:2203.04955*, 2022.

Anthony Hu, Gianluca Corrado, Nicolas Griffiths, Zachary Murez, Corina Gurau, Hudson Yeo, Alex Kendall, Roberto Cipolla, and Jamie Shotton. Model-based imitation learning for urban driving. *Advances in Neural Information Processing Systems*, 35:20703–20716, 2022.

Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. *Advances in Neural Information Processing Systems*, 37:819–844, 2024.

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *International conference on machine learning*, pp. 2688–2697. Pmlr, 2018.

Sven Kirchner, Nils Purschke, and Alois Knoll. A survey of world models for physical ai with uncertainty representation and control. *Discover Artificial Intelligence*, 6(1):1037, 2026.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International conference on machine learning*, pp. 5338–5348. Pmlr, 2020.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in neural information processing systems*, 33:11525–11538, 2020.

Thomas M. Moerland, Joost Broekens, Aske Plaat, and Catholijn M. Jonker. Model-based reinforcement learning: A survey. *Foundations and Trends in Machine Learning*, 16(1):1–118, 2023.

Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. *arXiv preprint arXiv:2209.00588*, 2022.

Chen Min, Dawei Zhao, Liang Xiao, Jian Zhao, Xinli Xu, Zheng Zhu, Lei Jin, Jianshu Li, Yulan Guo, Junliang Xing, et al. Driveworld: 4d pre-trained scene understanding via world models for autonomous driving. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15522–15533. IEEE, 2024.

Tuomas Oikarinen, Subhro Das, Lam M Nguyen, and Tsui-Wei Weng. Label-free concept bottleneck models. *arXiv preprint arXiv:2304.06129*, 2023.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14749–14759. IEEE, 2024.

Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. Slotformer: Unsupervised visual dynamics simulation with object-centric models. *arXiv preprint arXiv:2210.05861*, 2022.

Zhenya Yang, Zhe Liu, Yuxiang Lu, Liping Hou, Chenxuan Miao, Siyi Peng, Bailan Feng, Xiang Bai, and Hengshuang Zhao. Geniedrive: Towards physics-aware driving world model with 4d occupancy guided video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 35680–35690, 2026.

Mert Yuksekgonul, Maggie Wang, and James Zou. Post-hoc concept bottleneck models. *arXiv preprint arXiv:2205.15480*, 2022.

Wenzhao Zheng, Weiliang Chen, Yuanhui Huang, Borui Zhang, Yueqi Duan, and Jiwen Lu. Occworld: Learning a 3d occupancy world model for autonomous driving. In *European conference on computer vision*, pp. 55–72. Springer, 2024.

Yupeng Zheng, Pengxuan Yang, Zebin Xing, Qichao Zhang, Yuhang Zheng, Yinfeng Gao, Pengfei Li, Teng Zhang, Zhongpu Xia, Peng Jia, et al. World4drive: End-to-end autonomous driving via intention-aware physical latent world model. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 28632–28642. IEEE, 2025.

A CONCEPT ONTOLOGY

The CWM ontology combines directly annotated and derived variables describing ego motion, scene interactions, traffic control, environmental conditions, and navigation context. The complete definitions, derivations, and variable types are summarized in Table 4.

Table 4: CWM concept ontology constructed from Bench2Drive annotations and route metadata Jia et al. (2024). The ontology combines benchmark-provided attributes with spatially and temporally derived scene variables.

Variable	Definition	Source / Derivation	Type
Ego Speed	Current speed of the ego vehicle.	Direct ego-state annotation.	Continuous
Ego Acceleration	Rate of change of ego speed across consecutive observations.	$a_t = \mathrm{clip}\left(\frac{v_t - v_{t-1}}{\Delta t}, -15, 15\right)$.	Continuous
Lead-Vehicle Presence	Indicates whether a relevant vehicle is present ahead of the ego vehicle in the same lane.	Nearest forward same-lane vehicle within 60 m and ± 3.5 m laterally.	Binary
Lead-Vehicle Distance	Longitudinal separation between the ego vehicle and the selected lead vehicle.	Ego-relative longitudinal position of the selected lead vehicle.	Continuous
Lead-Vehicle Closing Speed	Rate at which the distance between the ego vehicle and the selected lead vehicle decreases.	$v_t^{\mathrm{close}} = \frac{d_{t-1} - d_t}{\Delta t}$, with temporal actor consistency.	Continuous
Time-to-Collision (TTC)	Estimated time to contact with the selected lead vehicle under the current relative closing rate.	$\mathrm{TTC}_t = \frac{d_t}{v_t^{\mathrm{close}}}$ for valid positive closing motion; capped at 20 s.	Continuous
Lead-Vehicle Braking	Indicates whether the selected lead vehicle is applying its brakes.	Brake state of the selected lead vehicle.	Binary
Pedestrian Presence	Indicates whether a pedestrian is present within the forward interaction region.	Pedestrian actor ahead of the ego vehicle within 40 m.	Binary
Pedestrian Distance	Longitudinal distance to the nearest relevant pedestrian.	Minimum ego-relative longitudinal distance among pedestrians satisfying the forward-corridor criterion.	Continuous
Nearby-Vehicle Count	Number of surrounding vehicles within the local traffic region.	Vehicle count within a 60 m radius, capped at 12.	Count
Cross-Traffic Presence	Indicates whether a nearby vehicle exhibits motion across the ego vehicle's direction of travel.	Relative geometry and orientation; crossing heading difference of 45°–135°.	Binary
Rear-Vehicle Presence	Indicates whether a vehicle occupies the rear corridor of the ego vehicle.	Vehicle behind the ego vehicle within 60 m and ± 3.5 m laterally.	Binary
Traffic-Light State	Current state of the traffic light governing the ego vehicle.	State of the nearest traffic light annotated as affecting the ego vehicle.	Categorical
Traffic-Light Distance	Distance from the ego vehicle to the relevant traffic light.	Distance to the nearest traffic light annotated as affecting the ego vehicle.	Continuous
Ego-Relevant Stop Sign	Indicates whether a stop sign currently applies to the ego vehicle.	Stop-sign annotation filtered according to its relevance to the ego vehicle.	Binary
Route Command	High-level navigation instruction associated with the current route state.	Route/navigation metadata.	Categorical
Weather Condition	Environmental weather configuration associated with the current driving scene.	Frame-level weather metadata.	Categorical

B CONCEPT GROUNDING ROLES

The concept ontology is partitioned according to its role in representation learning and evaluation. Thirteen atomic concepts provide direct supervision to the six grounded semantic factors, while *Lead-Vehicle Closing Speed*, *Cross-Traffic Presence*, and *Time-to-Collision* are excluded from direct concept supervision and retained as held-out semantic probes of the representation hierarchy. *Route Command* is treated separately as exogenous navigation context. The complete assignment of variables to these roles is summarized in Table 5.

Table 5: Concept roles in CGM. Thirteen atomic concepts directly supervise six grounded semantic factors, three concepts are retained as held-out semantic probes, and route command is treated as exogenous navigation context.

Role	Semantic Factor	Variables	No.
Atomic supervision	Ego Motion	Ego Speed, Ego Acceleration	2
Atomic supervision	Lead Vehicle	Lead-Vehicle Presence, Lead-Vehicle Distance, Lead-Vehicle Braking	3
Atomic supervision	Pedestrian	Pedestrian Presence, Pedestrian Distance	2
Atomic supervision	Traffic Control	Traffic-Light State, Traffic-Light Distance, Ego-Relevant Stop Sign	3
Atomic supervision	Surrounding Traffic	Nearby-Vehicle Count, Rear-Vehicle Presence	2
Atomic supervision	Environment	Weather Condition	1
Held-out probe	—	Lead-Vehicle Closing Speed, Cross-Traffic Presence, Time-to-Collision	3
Exogenous context	—	Route Command	1

C CONCEPTDRIVER: CONCEPT-SPACE PLANNING

The predictive dynamics learned by CWM can also support action optimization. ConceptDriver searches over future steering, throttle, and brake sequences through recursive CWM rollouts and evaluates their predicted concept trajectories using a concept-space objective:

$$\mathbf{A}^{*}_{t:t+H-1} = \arg\min_{\mathbf{A}_{t:t+H-1}} \sum_{h=1}^{H} \sum_{c \in \mathcal{C}_{\text{plan}}} w_c \, \ell_c\big(\hat{y}^{c}_{t+h}\big) \,, \tag{10}$$

where $\mathcal{C}_{\text{plan}}$ denotes the planning concepts, w_c their weights, and ℓ_c their cost functions. Optimization uses the Cross-Entropy Method (CEM), which iteratively samples candidate sequences and updates the sampling distribution toward lower-cost candidates.

Figure 5 presents an offline example over a 10-step horizon using 512 candidate sequences per CEM iteration and five iterations. Relative to the logged controls, ConceptDriver selects reduced throttle and stronger braking early in the horizon while modifying steering, producing a lower predicted ego-speed trajectory.

The model-predicted cost decreases from 84.19 for the logged sequence to 53.74 for ConceptDriver, a 36.17% reduction. This shows that recursive CWM rollouts can support action optimization according to predicted semantic consequences. Because optimization and evaluation both occur within the learned model, this demonstrates model-based planning capability rather than closed-loop driving improvement.

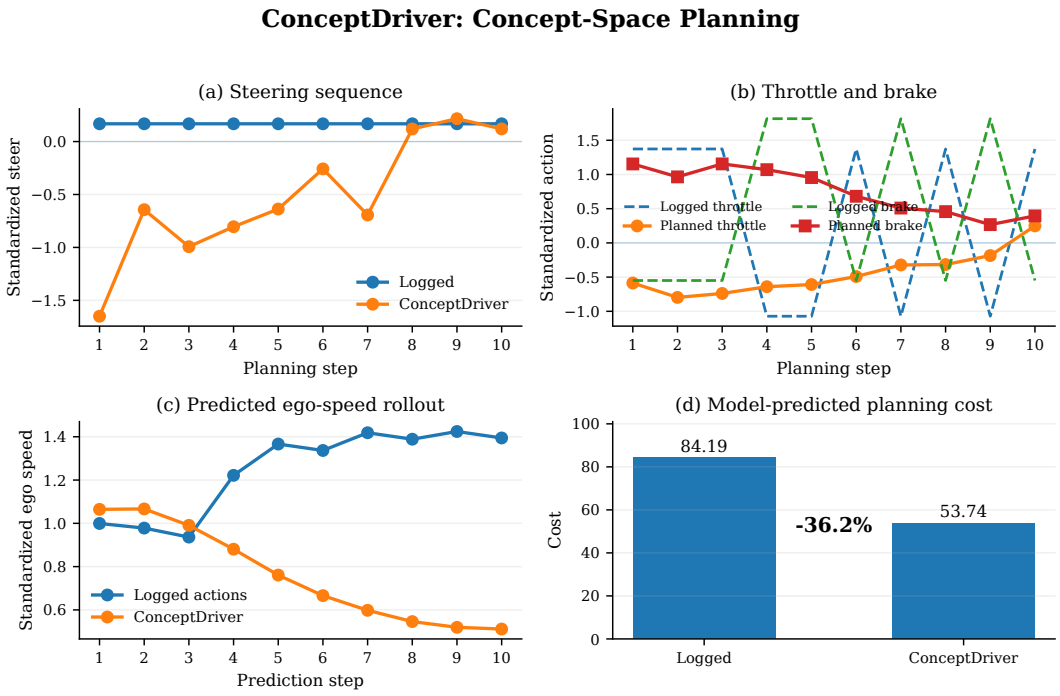


Figure 5: ConceptDriver planning through recursive CWM rollouts. (a) Logged and CEM-optimized steering. (b) Logged and optimized throttle and brake. (c) Predicted ego-speed trajectories. (d) Model-predicted planning cost, reduced from 84.19 to 53.74 (36.2%). Control and speed values are reported in standardized space.