World Motion Models: Flexible Sequence Modeling of SE(3) Trajectories

Jiahui Lei UC Berkeley
Qianqian Wang Harvard University
Trevor Darrell UC Berkeley
Angjoo Kanazawa UC Berkeley
{leijh,trevordarrell,kanazawa}@berkeley.edu qwang@seas.harvard.edu

Abstract

Equipping artificial agents with spatial intelligence requires a comprehensive generative prior over the dynamic 3D world. We propose World Motion Models (WMMs) that capture **"what was, is, and will be where across time"** via sparse *SE(3) pose trajectories*. WMMs are built on the observation that elements of dynamic scenes can be well approximated by a set of rigid SE(3) trajectories, a minimal yet expressive primitive for 4D modeling. This representation unifies articulated objects, human bodies, hand-object interactions, piecewise-rigid scene dynamics, camera motion, and even robot states and actions into a single shared space. Given this representation, we cast the joint distribution of these entities as a flexible sequence modeling problem, utilizing flow-matching with per-token noise levels. Coupled with a context token mechanism for non-sequential conditioning, this formulation supports any-to-any marginal conditioning across an arbitrary number of entities and time steps. Tasks such as future prediction, motion infilling, model-predictive control, inverse kinematics, cross-embodiment retargeting, and policy learning all reduce to the application of different masks over the same network. Experiments on 6 diverse applications of 3D vision and robotics demonstrate the versatility and flexibility of WMMs with strong performance.

https://jiahuilei.com/projects/wmm

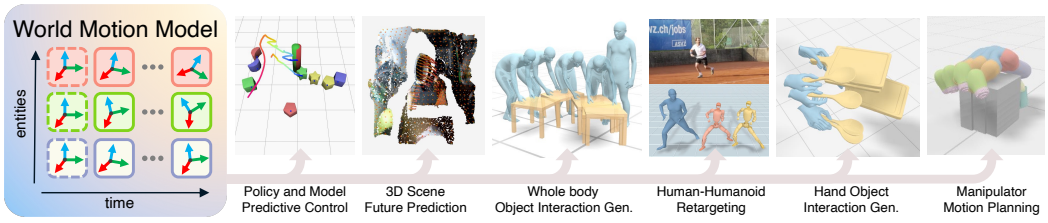

Figure 1: **World Motion Models** represent the 4D world across diverse entities and prediction modes in a unified manner, and can be flexibly conditioned for various downstream tasks.

1 Introduction

The world that surrounds us is inherently three-dimensional and highly dynamic—a 4D space where perception and interaction are fundamentally tied to movement. Physical actions are achieved by moving limbs or end-effectors, generating forces that translate into the motion of objects, while other agents dynamically navigate and act within this same environment. As humans, we naturally develop a strong prior over these moving entities: we intuitively anticipate how others will move, plan the physical motions required to achieve a desired goal, and foresee the physical consequences of our own actions. To act and reason in the physical world, intelligent systems benefit from the same

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

capability: a generative prior over the 4D world that captures the evolution of "what" and "where" across time.

In this work, we present a unified generative model which provides a general prior over the dynamic 3D world. Our approach is highly flexible with minimal assumptions and handles unstructured scenes and arbitrary object categories. By supporting various prompting modes, a single model enables diverse inference tasks ranging from world modeling (e.g., action-conditioned/free future prediction) to complex planning (e.g., motion infilling, inverse dynamics).

To build such a prior, we first need a unified representation that expresses the dynamic world in a physically grounded manner. Our insight is that complex, dense motion of the physical world can be compactly approximated by a sparse set of moving coordinate frames — specifically $SE(3)$ trajectories. Most entities of interest naturally reduce to this format: the dense flow of thousands of mesh vertices on a human or robot collapses into tens of joint trajectories [36, 60]; cameras and rigid objects track as single moving frames; and even non-rigid scene dynamics can be represented as interpolations of a small $SE(3)$ motion basis [43, 49, 85]. We therefore adopt the $SE(3)$ trajectory as our basic language for *what is where across time*. These $SE(3)$ trajectories subsume prior point trajectories in $\mathbb{R}^3$ but can be far sparser, replacing, for instance, the flow of thousands of mesh vertices with the tens of joint poses of a SMPL human [60] or a robot [36]. We further impose no category-specific structure: every entity is treated equally as a pose proxy, generalizing class-bound representations (SMPL [60], MANO [80], URDFs) to any unstructured scene. It is also orthogonal to video-based world modeling [3, 9, 70, 92, 103], which captures dynamics implicitly through color changes on a screen, entangling the underlying 3D motion with appearance, lighting, and viewpoint changes. In short, the $SE(3)$ trajectory provides a highly compact, physically grounded 3D abstraction for modeling arbitrary motion.

The question now becomes: how do we learn a generative prior over these $SE(3)$ trajectories? Since motion naturally unfolds over time, this is inherently a sequence modeling problem. Traditionally, this is treated as a strictly causal task: a standard world model or dynamics model predicts future states conditioned on past history and actions. However, humans can perform non-causal inference as well, such as determining the actions required to achieve a specific goal (inverse dynamics) or planning intermediate motions between waypoints. The inference type varies widely across applications. To support these diverse inference schemes, we propose World Motion Models (WMMs), flow-matching generative models with per-token noise levels. Inspired by Diffusion Forcing [15] and discrete diffusion [2, 69], WMMs treat each 6D $SE(3)$ pose at a given time step as an independent token, assign each its own noise level, and denoise the entire token field in a single shared forward pass.

Once trained, a WMM admits inference over arbitrary numbers of entities and time steps simply by dictating which tokens are clean (the condition) and which are noised (the prediction). Given past tokens, it acts as a future predictor $p(s'|s)$; applied to action tokens, it becomes a feed-forward policy $p(a|s)$; and when both past states and actions are provided, it acts as an action-conditioned dynamics model $p(s'|s, a)$ supporting model-predictive control. Ultimately, WMMs realize any $p(\text{unknown}|\text{known})$. By simply changing what is known, the exact same network transitions seamlessly between serving as a policy, an action prior, and a world model.

While standard trajectory modeling assumes a strictly consecutive temporal horizon, many practical tasks require reasoning beyond it. For example, planning a complex interaction often relies on an abstract goal state situated arbitrarily far in the future. Similarly, tasks like cross-embodiment retargeting require inferring correspondences between trajectories, such as those of a human and a robot. Providing a few paired demonstration frames from outside the current episode serves as an in-context template that tells the model "what is what." To support these non-consecutive anchors, we enhance the flexibility of WMMs by introducing a set of *context* $SE(3)$ *tokens*. Rather than being bound to the continuous physical timeline of the current episode, they are assigned to a decoupled context timeline with its own learnable positional embeddings. Mechanically, these context tokens are treated identically to real trajectory tokens, meaning they can seamlessly condition the model or be generated themselves.

Together, this general $SE(3)$ formulation and our flexible sequence modeling enable WMMs to demonstrate broad versatility across applications, including robot policies and model-predictive control (MPC), scene future prediction, human-object interaction generation, and multi-embodiment humanoid retargeting (Sec. 3.1), all with SOTA performance.

Contributions. We propose a general formulation that uses the sparse $SE(3)$ trajectory as a unified representation for 4D world modeling. To learn a flexible generative prior over this space, we introduce WMMs, flexible flow-matching models with per-token noise levels. By supporting any-to-any marginal conditioning alongside a context token mechanism, WMMs deliver strong performance across diverse 3D vision and robotics tasks.

2 World Motion Models

2.1 Representation: A Language for Motion Modeling

World Motion Models aim to answer **what was, is, and will be where** across time. The thread connecting an entity at t_1 with the same entity at t_2 is its motion, and motion in real-world scenes is largely low-rank: a complex scene is driven by a relatively small set of *eigen* rigid trajectories, while dense spatial deformation arises as a local combination of those bases [43, 49, 85]. We assume the dynamic 3D world can be approximated by a finite set of rigid $\mathrm{SE}(3)$ trajectories, with dense spatial deformation recovered by interpolation, skinning, or binary assignment. $\mathrm{SE}(3)$ pose trajectories serve as the language with which we model the dynamic world. We leave modeling general linear local motion (scaling, shearing) to future work.

Coverage. As Fig. 2-(A) shows, this primitive approximates the motion of the principal subjects of 3D vision and robotics. *Scene future prediction* over 3D point trajectories [36, 46, 48] is the rotation-free special case of $\mathrm{SE}(3)$ trajectories. *Human and hand-object motion* fits naturally: parametric body and hand models [47, 60, 80, 106] are sets of bone-joint $\mathrm{SE}(3)$ frames, and articulated parts contribute additional rigid trajectories with per-part local shape. *Robot kinematics* treats each link as one rigid trajectory plus its mesh, the substrate for motion planning, retargeting, and control. *Robot state and action* are equally expressible whenever the controller (PD gains, etc.) is fixed, since the action then reduces to a set of virtual target poses. Thus, the same language captures both the observation and action of a standard control problem. *Camera motion* is, by definition, an $\mathrm{SE}(3)$ trajectory of the viewpoint.

A pose at a time as a token. The elementary unit of our representation is a continuous token of "where", given by a pose $T^{(l)} \in SE(3)$ and paired with a feature $F^{(l)} \in \mathbb{R}^D$ that supplies the "what" at time l. From now on we use $\{i, j, k\}$ to index entities and $\{l, m, n\}$ to index successive time steps. The feature is constant across time (shared along the trajectory) and is composed as $F_i = [\, f_i^{\text{sem}};\, f_i^{\text{geo}}\,]$: the semantic part f_i^{sem} is either a DINO pixel feature [71] for video-prompted entities or a learnable embedding for coarse semantic categories (e.g. a body joint vs. an object). The

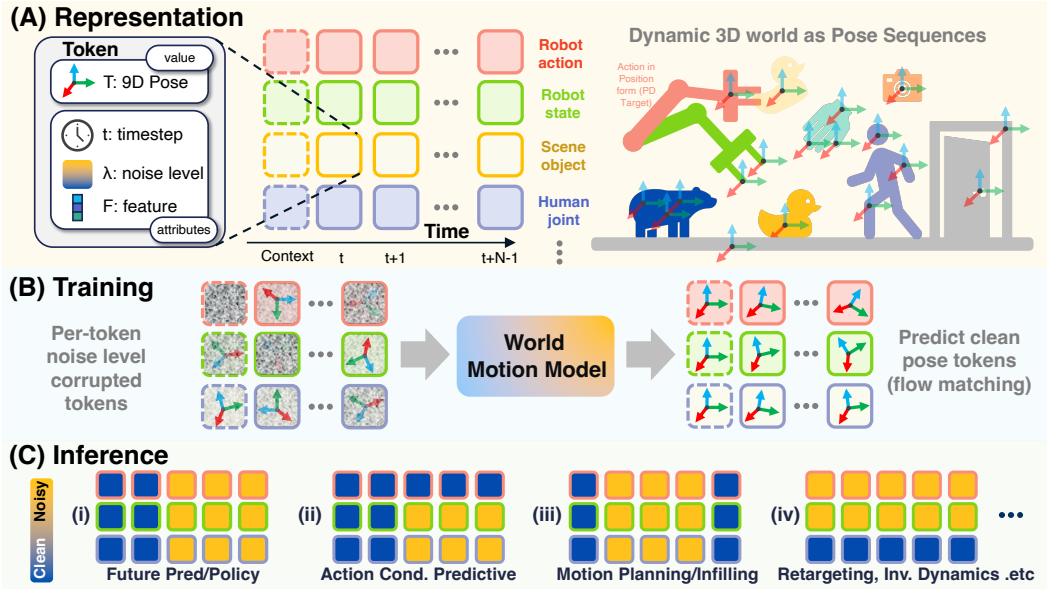

Figure 2: **World Motion Models**: (A) Dynamic scenes can be approximated by pose sequences. (B) During training, tokens corrupted at per-token noise levels are denoised to clean tokens. (C) During inference, various conditioning masks can be applied to one trained model.

geometric part f_i^{geo} is a small descriptor extracted by a PointNet [76] over the local point cloud when available (e.g. the local cloud of the object in human-object interaction or the shape of a robot body link) and is omitted otherwise (e.g. for a point prompt from a video). More details and ablations are in Appendix A.2. The pose token $T_i^{(l)}$ itself is parameterized as a 9-vector built from three points in $\mathbb{R}^3$,

$$T_i^{(l)} \;\equiv\; \big(\mathbf{x},\;\; \mathbf{x}+\delta\,\mathbf{r}_1,\;\; \mathbf{x}+\delta\,\mathbf{r}_2\big) \in \mathbb{R}^9, \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^3$ is the position, $(\mathbf{r}_1, \mathbf{r}_2)$ are the first two columns of the rotation matrix under the 6D parameterization of Zhou et al. [104], and δ is a small fixed scalar (e.g. 0.2) empirically chosen to normalize the primal diffusion signal to a standard normal distribution (Appendix A.1). This 9-D form folds the unit-vector pair $(\mathbf{r}_1, \mathbf{r}_2)$ into the same $\mathbb{R}^3$ space as the translation, making normalization and training easier. Ground-truth labels for these poses at different time steps are widely available: they are read directly from parametric human reconstructions [27, 73] or from robot action recordings. For real videos, they can be computed using a closed-form approximation by defining a proper neighborhood around point traces produced by recent 4D reconstruction tools [36, 46, 48].

Dynamic scenes as trajectory sequence sets. An entity i moving through time produces a sequence of such pose tokens decorated with the same feature,

$$\tau_i \;=\; \big([\,T_i^{(l)},\, T_i^{(m)},\, T_i^{(n)},\, \ldots];\; F_i\,\big), \tag{2}$$

and a dynamic 3D scene is the set of M such trajectories,

$$\mathcal{X} \;=\; \{\tau_i,\, \tau_j,\, \tau_k,\, \ldots\} \;=\; \big\{\,([T_i^{(l)}, T_i^{(m)}, T_i^{(n)}, \ldots]; F_i)\big\}_{i=1}^{M}, \tag{3}$$

which we abbreviate as $\mathcal{X} = (\mathcal{T}, \mathcal{F})$ with the union of pose-sequences $\mathcal{T} = \{[T_i^{(l)}, T_i^{(m)}, T_i^{(n)}, \ldots]\}_{i=1}^{M}$ and the corresponding feature set $\mathcal{F} = \{F_i, F_j, F_k, \ldots\}$. Any dynamic scene over a causal temporal window can be expressed as such a pose-sequence set. However, many problems also benefit from non-causal context, e.g. a goal state far beyond the rollout window for an MPC controller to optimize toward, or a handful of motion-pair exemplars that specify a cross-embodiment retargeting relationship between sequences. We therefore extend each trajectory with a prefix of virtual time steps disjoint from the causal window,

$$\tau_i \;=\; \big([\,\tilde{T}_i^{(a)},\, \tilde{T}_i^{(b)},\, \ldots \;\big|\; T_i^{(l)},\, T_i^{(m)},\, T_i^{(n)},\, \ldots];\; F_i\big), \tag{4}$$

where $a, b, \ldots$ are virtual indices distinguished by a separate positional embedding (Sec. 2.4). Context tokens participate in training and inference exactly as causal ones, so a goal, an exemplar, or a label condition is simply another set of tokens. Without loss of generality, τ refers to this augmented form throughout, and Sec. 3 instantiates several choices of what to place in the virtual context.

Modeling target. A vanilla generative prior over this language captures $P\big(\,\mathcal{T}\,\big|\,\mathcal{F},\,\mathcal{A}\,\big)$, where $\mathcal{A}$ holds auxiliary conditions such as language or image tokens and $\mathcal{F}$ holds the per-token "what" decoration. Real applications, however, condition on arbitrary partial state rather than on $\mathcal{F}$ and $\mathcal{A}$ alone, so we instead study the family of any-marginal conditionals

$$P\big(\,\mathcal{T}_{\text{unk}}\,\big|\,\mathcal{T}_{\text{obs}},\,\mathcal{F},\,\mathcal{A}\,\big), \qquad \forall\,\mathcal{T}_{\text{obs}} \subset \mathcal{T},\;\; \mathcal{T}_{\text{unk}} = \mathcal{T} \setminus \mathcal{T}_{\text{obs}}, \tag{5}$$

i.e., every distribution induced by partitioning the trajectory set into an observed subset and its complement. We treat this any-to-any property as the operational definition of a world model, and a single WMM supports it.

2.2 Training: Flow Matching with Per-Token Noise Level

We learn the any-marginal prior of Sec. 2.1 with a single flow-matching model under *asynchronous* noise levels (Fig. 2-(B)): each token (one entity at one time step) is assigned its own scalar level, sampled i.i.d. This is inspired by Diffusion Forcing [15] and diffusion large language models (dLLMs) [69, 93], which can model sequences in arbitrary order and over arbitrary subsets.

Forward process. Given a clean trajectory set $\mathcal{T}$ with M entities and N time steps (with pose tensor shape $M \times N \times 9$), we sample a per-token noise-level field $\Lambda \in [0,1]^{M\times N}$ with each entry $\lambda_{i,l} \sim \mathcal{U}[0,1]$ i.i.d., together with a Gaussian noise field $\epsilon \in \mathbb{R}^{M\times N\times 9}$, $\epsilon_{i,l} \sim \mathcal{N}(0, I_9)$. Each token moves along the path toward its sampled noise, with $\lambda_{i,l}$ broadcast across all 9 channels, giving the noised pose

$$T_i^{(l)}\Big|_{\lambda_{i,l}} \;=\; (1-\lambda_{i,l})\,T_i^{(l)} \;+\; \lambda_{i,l}\,\epsilon_{i,l}. \tag{6}$$

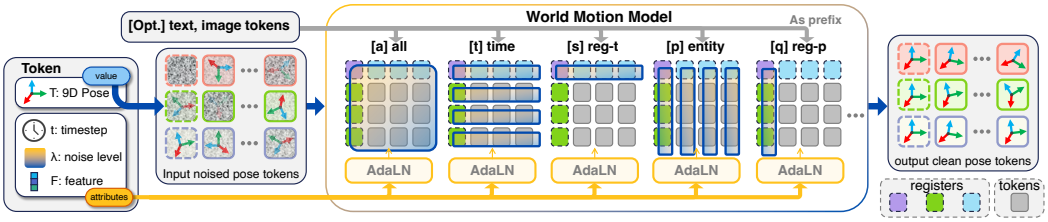

Figure 3: **Network**: The WMM architecture contains five factored attention types over a token grid padded with three register blocks, enabling efficient modeling (Sec. 2.4).

Velocity target and loss. Our model $\mathcal{M}_\theta$ is a velocity field in the flow-matching sense [54, 57] that predicts the velocity from the clean pose toward its sampled noise. The interpolation target is $v^\star_{i,l} = \epsilon_{i,l} - T^{(l)}_i$, and, conditioned on the noised trajectory from Eq. 6, the feature set $\mathcal{F}$, the auxiliary condition $\mathcal{A}$, and the noise-level field Λ, the network is trained with a per-token error,

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathcal{T},\Lambda,\epsilon} \sum_{i,l} \left\| \mathcal{M}_\theta(\mathcal{T}_\Lambda, \mathcal{F}, \mathcal{A}, \Lambda)_{i,l} - v^\star_{i,l} \right\|_2^2. \tag{7}$$

Why this single objective is enough. Eq. (7) bakes the any-marginal prior of Sec. 2.1 into one network [15, 93]: each random Λ splits tokens into nearly-clean (observation-like) and nearly-noised (target-like), covering every conditional reverse process, with the deterministic observed/target split as the limiting case. Fixing Λ at inference (Sec. 2.3) samples the chosen marginal. The noise level scheduling can also be biased toward deployment patterns, e.g. more often scheduling a small λ on history and a large one on the future to sharpen forecasting while preserving coverage during training.

2.3 Inference: Various Marginal Distributions from the Same WMMs

A trained WMM is an entire family of conditional samplers under one set of weights. At inference, the user picks a per-token mask of observed and target tokens, sets $\lambda_{i,l} = 0$ on the former and $\lambda_{i,l} = 1$ on the latter, and integrates $\mathcal{M}_\theta$ from $\lambda = 1$ down to 0: observed tokens stay as input, while target tokens are transported by the velocity field, yielding a sample from $P(\mathcal{T}_{\text{unk}} \mid \mathcal{T}_{\text{obs}}, \mathcal{F}, \mathcal{A})$. Different masks cover various applications (Fig. 2-C), for example:

(i) Future prediction and policy. $\lambda_{i,l} = 0$ for $l \le l_0$, $\lambda_{i,l} = 1$ for $l > l_0$. State-only entities give future prediction; with action entities in $\mathcal{T}$, the same mask is a robot policy.
(ii) Action-conditioned predictive model (forward dynamics). Action entities known, future states unknown: a learned forward simulator for model-predictive control.
(iii) Motion planning and in-filling. Sparse keyframes (possibly partial) are observed; the rest are targets: robot motion planning and motion interpolation.
(iv) Retargeting, partner prediction, and inverse dynamics. Partition by entity: $\lambda_{i,l} = 0$ for $i \in \mathcal{O}$, $\lambda_{i,l} = 1$ otherwise. Taking $\mathcal{O}$ to be one body gives cross-embodiment retargeting; taking it to be a hand gives hand-object HOI partner prediction (or vice versa); taking it to be the state entities gives inverse dynamics.

Note that any known–unknown pattern is supported, beyond the examples mentioned above. Such a design has additional benefits: **robustness to observation noise.** Real observations are imperfect, and WMMs accommodate this by setting $\lambda_{i,l}$ on observed tokens to a small positive value (e.g. 0.05) rather than exactly 0, which turns each observation into a strong hint and lets the integrator slightly rectify the token along the velocity field as the noise level decreases. **Generalization to many sequences:** RoPE in a transformer already helps the model generalize beyond the number of sequences seen during training. A simple inference-time algorithm (App. B.2) extends this generalization further when combined with the per-token noise formulation, enabling dense trajectory generation well beyond the training count.

2.4 Implementation: Model Architecture

As shown in Fig. 3, our model supports arbitrary numbers of entities and time steps while addressing three issues: (i) *per-token noise levels*: $\lambda_{i,l}$ has to be specified *per token*, not as a single global scalar as in a conventional flow-matching backbone [54, 74]; (ii) *scale*: a modest 30-frame, 400-trajectory scene already contains 12K tokens, on which dense self-attention is expensive; (iii) *missing*

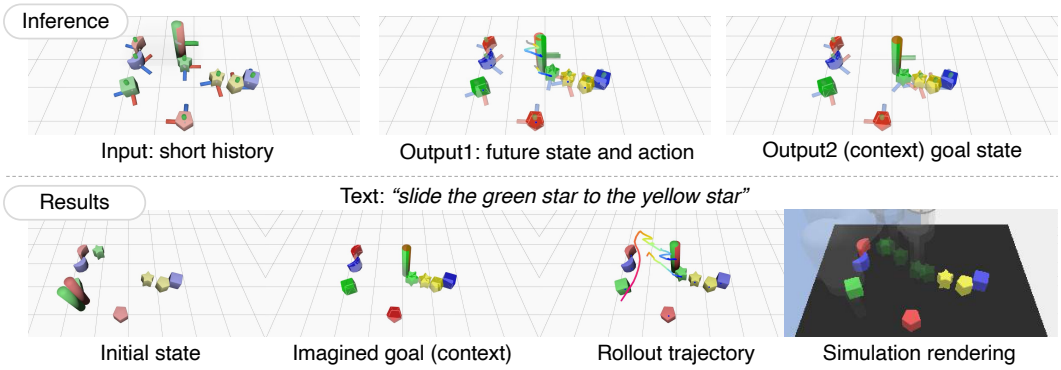

Figure 4: **Language Table.** Top: inference pattern; history is the input, while future and goal are predicted by WMMs. Bottom: policy rollout results.

Table 1: **Language Table.** Per-task and macro-average success rate (%).

Method	B2B↑	B2AL↑	B2BRL↑	B2RL↑	Sep↑	Avg↑
Scratch [94]	4.0	6.0	10.0	6.0	52.0	15.6
UniPi [24]	14.0	4.0	12.0	10.0	72.0	22.4
VPT [6]	36.0	38.0	48.0	26.0	70.0	43.6
LAPA [94]	58.0	56.0	52.0	48.0	96.0	62.0
ActionVLA [94]	76.0	72.0	76.0	70.0	90.0	76.8
LCB [81]	**90.0**	71.0	66.0	73.0	99.0	80.0
LangTable orig. [62]	88.0	65.0	70.0	68.0	94.0	77.0
WMM	**90.0**	**76.0**	**80.0**	**90.0**	**100.0**	**87.2**

Table 2: **Plain Policy versus MPC.** On Language Table absolute tasks.

	B2B		B2AL	
	Plain	MPC	Plain	MPC
Succ.↑	90.0	**100.0**	76.0	**80.0**
Mean steps↓	79.5	**79.1**	92.7	**87.2**
Med. steps↓	73.0	**68.5**	87.0	**85.5**
Min steps↓	26	**16**	14	**2**
Max steps↓	198	**175**	187	**171**

observations: real data come with invalid intervals (occlusion, truncation, tracker drop-out) that cannot simply be masked during training. We address each as follows: **(1) Per-token conditioning via AdaLN.** For each token (i, l) we concatenate (a) a sinusoidal embedding of $\lambda_{i,l}$, (b) the feature F, and (c) a pooled embedding of the auxiliary tokens $\mathcal{A}$ (e.g. a language summary). A per-block wide multi-layer FFN maps this vector to per-token AdaLN scale-and-shift parameters that modulate every transformer block [74]. **(2) Registers and factored attention.** We pad the $M \times N$ token grid with three blocks of learnable registers [21] and apply five attention types on the padded grid: `a` self-attends across all tokens; `t` and `p` are time- and entity-axis self-attention; `s` and `q` are register-only counterparts on the same two axes, widened aggressively in channels at negligible runtime cost to add expressivity. **(3) Training under missing observations.** Invalid time steps must still participate in attention, since at inference the missing mask is unknown. We pin invalid tokens to $\lambda_{i,l} = 1$ (pure noise) and sum the loss in Eq. (7) only over tokens with ground-truth labels. Virtual-context tokens (Sec. 2.1) are flagged by a 1-bit prefix mask and given learnable position embeddings. Text or image tokens attach as an attention prefix.

3 Experiments

We evaluate WMMs on open-source benchmarks spanning policy learning, future prediction, HOI, and retargeting, plus a set of ablations. We train a separate WMM checkpoint on each benchmark's training split; joint training of one WMM across all benchmarks is left to future work. The entities modeled by a WMM are defined according to the task: examples include a link or joint of a robot, human body, or object (Secs. 3.1, 3.4, and 3.3), or any prompted query point in the general background (Sec. 3.2); see below for details.

3.1 Language Table: Policy Learning and Forward Dynamics

Language Table [62] is a tabletop pushing benchmark with five task families. We follow the evaluation protocol of LAPA [94] and LCB [81]: 50 rollouts per family on the eight-block setup with a 200-step horizon, reporting per-family and macro-average success rates. **Setup.** As in Fig. 4, the entity set comprises the 6D pose of every block together with the robot end-effector state (green) and action target (red); the context tokens are configured to capture the final frame of a successful trajectory. The feature descriptor of each token is defined as follows: F_{sem} consists of three learnable embeddings

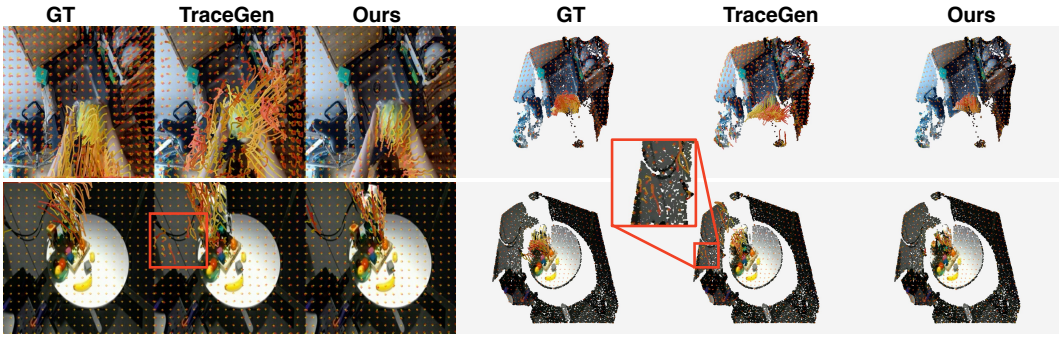

Figure 5: **TraceGen.** Qualitative results on the TraceGen benchmark [46].

Table 3: **TraceGen.** 3D-trajectory metrics on the Droid and EpicKitchen test splits.

Method	Dr MSE↓	Dr MAE↓	Dr Endp.↓	Ep MSE↓	Ep MAE↓	Ep Endp.↓
TraceGen [46]	0.206	1.289	0.285	0.445	2.721	0.791
WMM, per-dataset (P=128)	**0.151**	**1.129**	**0.200**	**0.322**	**2.130**	**0.556**
WMM, per-dataset (P=400)	0.178	1.169	0.236	0.329	2.167	0.580
WMM-Joint (P=128)	0.167	1.172	0.225	0.346	2.256	0.615
WMM-Joint (P=400)	0.187	1.179	0.259	0.351	2.287	0.621

for object, link-state, and link-action, while F_{geo} is obtained by applying a PointNet to the local point cloud of the object or link. We train one model on the public oracle trajectories across all five tasks, and inference runs in the policy mode of Sec. 2.3: given a five-step history and language prompt, a ten-step action chunk is read out from the action entity in the future prediction, ensembled across queries as in ACT [100]. The context (goal state) is also generated from the language and observation. **Results.** As in Tab. 1 and Fig. 4, a WMM obtains the strongest macro-average, demonstrating that WMMs are effective and accurate as policies.

Action optimization with MPC. WMMs can also be used as a differentiable world model rather than a plain policy. For tasks with absolute-position instructions (B2B, B2AL) in LangTable, given the generated goal state (via context tokens), we can apply model predictive control (MPC) to optimize the predicted action so as to maximize a heuristically defined value function: the similarity between the final frame of the action chunk and the generated goal. Since our model is fully differentiable, gradient-based MPC can be hybridized with sampling-based methods (full procedure in App. B.3); Tab. 2 compares the plain policy with the MPC-optimized policy on B2B and B2AL, indicating both a higher success rate and faster rollout to success.

3.2 TraceGen: Dense 3D Scene Future Prediction

TraceGen [46] is a 3D trajectory generation benchmark in which the model takes a single RGB-D start frame and a language instruction and predicts the motion of 400 keypoints over 32 steps. We follow its protocol on the two subsets that provide 3D trajectories, Droid and EpicKitchen, reporting the trajectory-generation errors MSE, MAE, and endpoint MSE (each $\times 100$). **Setup.** Each of the 400 keypoints is one entity carrying a 3D position and an orientation (computed by local Procrustes in the dataset), starting at the first frame (t=0). We train four WMM checkpoints over the experimental space (Droid only, EpicKitchen only, and joint on both) at two keypoint window sizes (P=128 and P=400, where P is the number of trajectories seen during training). No special context token is used in this experiment. **Results.** As shown in Tab. 3 and Fig. 5, WMM achieves the best performance in every column in both environments. Note that the $P = 128$ variant is trained with 128 randomly selected trajectories, meaning that it can accept random queries during inference. Although only 128 trajectories are observed in parallel during training, the model can directly infer all 400 trajectories and even achieves better performance (P=128 wins all metrics); we hypothesize that this stems from the random sampling of 128 out of 400 trajectories during training, which prevents overfitting. The experiments also show that joint training across both domains is feasible, suggesting that future work can pursue a single unified model across domains.

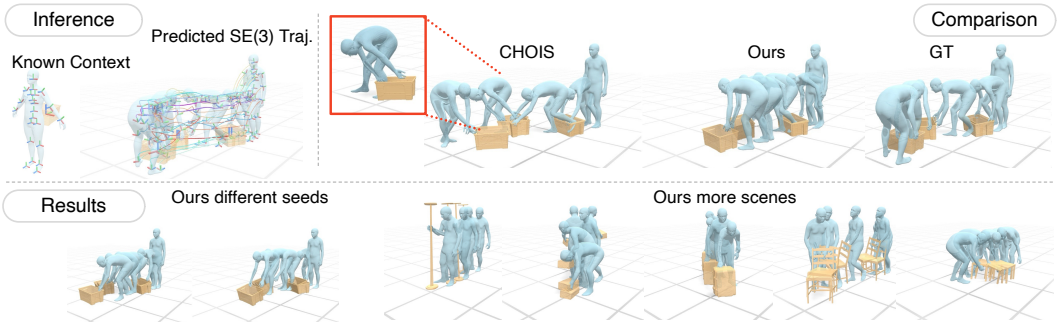

Figure 6: **OMOMO.** Qualitative results on OMOMO [50]. Top left: inference pattern. Top right: comparison with CHOIS [51]. Bottom: additional results from WMMs.

Table 4: **OMOMO.** Results under the CHOIS [51] protocol.

Method	$T_s \downarrow$	$T_e \downarrow$	$T_{xy} \downarrow$	$H_{feet} \downarrow$	FS$\downarrow$	$C_{prec} \uparrow$	$C_{rec} \uparrow$	$C_{F1} \uparrow$	$C_{\%} \uparrow$	MPJPE$\downarrow$	$T_{root} \downarrow$	$T_{obj} \downarrow$	$O_{obj} \downarrow$
InterDiff [90]	–	158.84	72.72	0.90	0.42	0.63	0.28	0.33	0.27	25.91	63.44	88.35	1.65
MDM [83]	5.18	33.07	19.42	6.72	0.48	0.72	0.47	0.53	0.43	17.86	34.16	24.46	1.85
Pred-OMOMO [50]	2.39	8.03	4.15	7.08	0.40	0.73	0.66	0.66	**0.62**	18.66	28.39	16.36	1.05
CHOIS w/o L_{obj} [51]	5.76	14.16	8.44	6.55	0.40	0.75	0.50	0.55	0.43	14.34	21.97	15.53	**0.98**
CHOIS w/o F_{all} [51]	1.75	6.61	**2.69**	6.64	0.38	0.78	0.49	0.55	0.41	15.23	24.13	**11.51**	0.99
CHOIS [51]	1.71	**6.31**	2.87	4.20	**0.35**	**0.80**	0.64	0.67	0.54	15.30	24.43	12.53	0.99
WMM	**0.15**	6.48	4.40	**0.84**	0.56	0.79	**0.67**	**0.76**	0.59	**12.07**	**19.87**	16.53	1.03

3.3 OMOMO: Human-Object Interaction

OMOMO [50] is a benchmark for object-conditioned full-body human-object interaction synthesis: given a partial object keyframe and a language prompt, the model generates the body and object motion. We follow the protocol of CHOIS [51]. **Setup.** The entity set comprises the SMPL body joints and the interaction object's rigid poses. The Context tokens are always set to a SMPL human in an A-pose (which conveys entity identities) and the object at the origin. The time window is set to the dataset's full length of 120 frames.

Note that in this human body example and the following humanoid and hand experiments, WMM directly models pose trajectories even when a structure such as the SMPL model is known, enabling WMM to model general and diverse morphologies within a single architecture and model. The following visualizations are produced by directly animating the meshes with linear blend skinning (LBS) or rigid transformations using the poses predicted by WMM, showing the accuracy of our model. To conduct a fair comparison (e.g., baselines use SMPL parameters to compute metrics), we convert the output of WMM to the corresponding parametric space; further discussion of this projection is provided in Appendix A.3.

Results. Tab. 4 and Fig. 6 show comparable or even better performance on the HOI synthesis task. Fig. 6 further shows that when CHOIS produces implausible predictions in which the human intersects the object, the WMM maintains plausibility and can generate diverse motions. Additionally, to compare model size and efficiency, we measure the parameter counts of TraceGen and WMM (674.6M vs 438.8M) and their inference times for 400 tracks over 32 steps (2.66 s vs 4.18 s).

3.4 Human and Humanoid: Kinematic Retargeting

We construct a benchmark that takes a SMPL human motion sequence and produces a Unitree G1 humanoid [84] trajectory. We evaluate retargeting quality by feeding the trajectory to a low-level robot controller (TWIST [97]), rolling it out in physics simulation, and measuring tracking error and success rate. Three input settings probe WMMs' flexibility: clean human input at 30 FPS; sparse keyframes at 3 FPS that the model must infill; and human input with noise at every frame (σ=0.1 on axis-angle and root translation). **Setup.** The entity set comprises the SMPL human and the G1 humanoid links, paired across embodiments inside the Context-token exemplar frames (Sec. 2.1; two samples in Fig. 7). A WMM is trained on a mix of AMASS [64] and Embody3D [66]; a single checkpoint serves all three settings. The humanoid's kinematic reference (ground truth) is obtained by running and filtering PHUMA [45], a slow offline optimization method. **Results.** Tab. 5 and Fig. 7

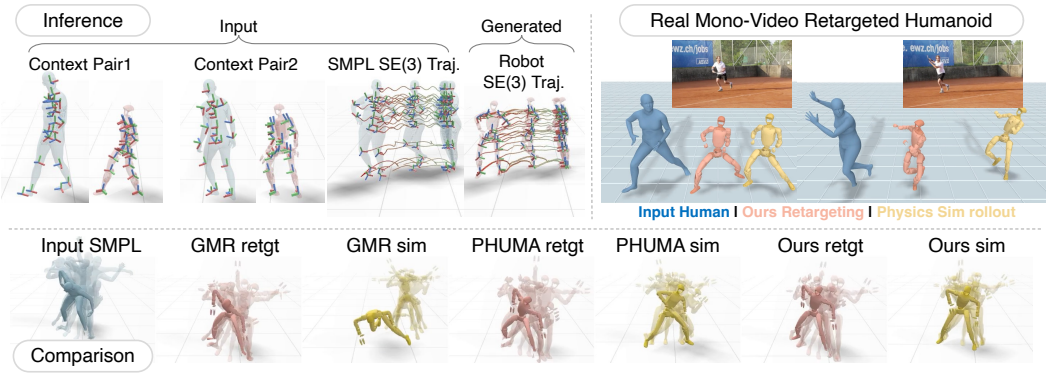

Figure 7: **Human-to-Humanoid Retargeting.** Top left: inference pattern. Top right: retargeting from real-world video. Bottom: comparison against baselines.

Table 5: **Humanoid Retargeting.** Results under three conditioning regimes (clean, sparse-keyframe in-fill, per-frame noised).

	Clean (30 FPS)				In-fill (3 FPS keyframes)			Noisy (σ=0.1)		
Method	FPS↑	SR-seq↑	gerr↓	lerr↓	SR-seq↑	gerr↓	lerr↓	SR-seq↑	gerr↓	lerr↓
GMR [1]	35.03	0.7945	0.6911	0.0651	0.8095	0.6542	0.0592	0.7160	1.0196	0.1158
PHUMA [45]	2.27	0.8255	**0.5509**	0.0530	0.6535	1.3744	0.0745	0.8060	**0.7269**	0.0703
WMM	**86.97**	**0.8690**	0.5938	**0.0420**	**0.8525**	**0.6371**	**0.0518**	**0.8530**	0.7785	**0.0691**

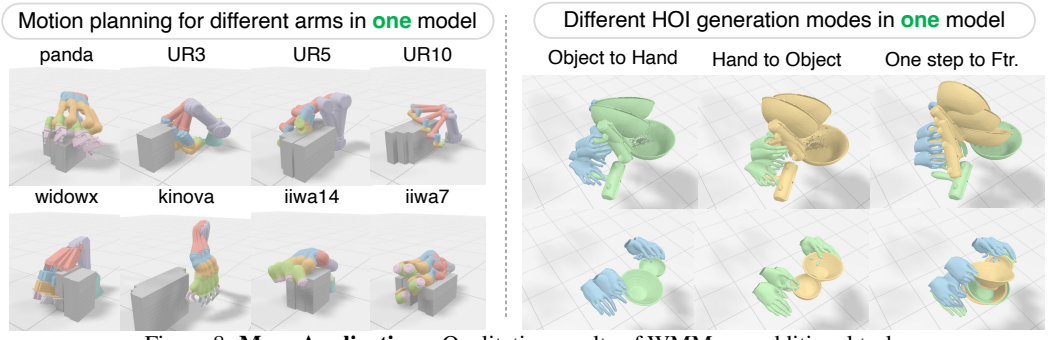

Figure 8: **More Applications.** Qualitative results of WMMs on additional tasks.

report the comparison against GMR [1] and PHUMA [45]. GMR is a fast, real-time method but provides limited quality; PHUMA is an offline optimization method that produces more physically plausible results (less overlap, etc.) but is slow. The resulting WMM is a retargeting prior that is fast (note the throughput reported in the first column) and learns a prior over physically plausible, safe motion across sequences, providing higher-quality and safer retargeting motion. Fig. 7 also shows real-world monocular-video-based humanoid retargeting and control results.

3.5 More Applications

The left side of Fig. 8 shows an additional application of WMMs: motion planning with known first and last timesteps (Fig. 2-(C ii), labelled by PyRoKi [42]). This works across more than 8 different manipulators within a single model, showing that our flexible modeling techniques can model diverse kinematic targets across morphologies. The right side of Fig. 8 shows hand-object interaction generation on TACO [58] under various known-unknown patterns within one model (the green mesh denotes the known condition), showing that WMM can be used in diverse ways during inference.

3.6 Ablations

We ablate per-block AdaLN and the per-token-noise schedule (Sec. 2.2) on Language Table B2B, and architectural primitives (registers, factored axis-wise attention; Sec. 2.4) on TraceGen EpicKitchen. Each ablation toggles one field; details in App. B.4. **Language Table.** Tab. 6 (left): removing AdaLN

drops B2B success by 6 pp but the model still learns a competent policy; removing per-token noise (using synced noise with guidance-based conditioning) yields no successful rollout, since policy mode requires very high accuracy. **TraceGen.** Tab. 6 (right): collapsing the middle three layers to plain DiT (no efficient register) degrades performance because the same hardware then supports a smaller batch size and slower training. Collapsing all seven to plain DiT (no axis-wise factorization) yields a $\sim 15\times$ MAE regression, as the model loses the explicit temporal–spatial inductive bias.

4 Related Work

Motion Generation and World Models. Existing generative motion models are fragmented by actor class through morphology-specific parameterizations [41, 53, 83], while general unified approaches [36, 48, 70] operate in dense observation spaces such as pixel-aligned point tracks, 3D flow, or video entangling 3D dynamics with appearance, or solely in 2D pixel motion space [68, 77, 78]. A piecewise-rigid lineage uses $SE(3)$ transforms [10, 33] but is restricted to single-agent or fixed-object settings. WMMs instead encode every moving unit (robot link, body part, scene primitive) as a (sparse) set of $SE(3)$ tokens in a shared abstraction, leading to a potentially more unified and shared motion prior over all moving subjects in the physical world.

Table 6: **Ablations.** Language Table B2B (left), TraceGen EpicKitchen (right).

Variant	SR↑	Variant	MAE↓	MSE↓	Endp.↓
WMM	**0.860**	**WMM**	**2.167**	**0.329**	**0.580**
no AdaLN	0.800	no eff.	2.356	0.344	0.602
no per-tk	0.000	plain DiT	32.05	16.97	17.64

Flexible Policies and Sequence Modeling. Recent work couples world modeling with policy via video–action models [52, 56, 105] or latent-dynamics planners [28, 29], but ties the world model to a fixed morphology or pixel tokens. Diffusion Forcing [15] and discrete masked diffusion [2] introduce per-token noise schedules for flexible conditioning. Specifically, Diffusion Forcing [15] is the modeling method closest to ours. Although theoretically general, it primarily focuses on temporal scheduling across sequence frames. Our method generalizes the idea of per-token noise to spatial and temporal axes, and our architectural design makes this practical for more than 10K tokens. We provide a unified representation of heterogeneous SE(3) trajectories and add a decoupled context-token mechanism, so one network spans prediction, policy rollout, and non-causal planning. See App. B.1 for an extended discussion.

5 Conclusion

We propose World Motion Models, a unified generative prior over the dynamic 3D world via sparse $\mathrm{SE}(3)$ pose trajectories. A per-token-noise-level flow-matching model with context tokens supports any-marginal conditional inference from one network, attaining state-of-the-art results on policy learning, scene future prediction, human-object interaction, and humanoid retargeting.

Limitations. **(i)** We train one WMM checkpoint per benchmark; joint training across domains and increasing the amount of supervision data are left to future work. **(ii)** Generalizing $SE(3)$ to more general linear groups is also a future direction. **(iii)** The current framework requires geometry (e.g., depth and estimated current and past states) as input; relaxing these requirements and enabling direct inference from RGB videos are also directions for future work. **(iv)** Another limitation of the current model is that it does not generalize to novel morphologies unseen during training; further investigation of in-context learning with our proposed context mechanism may suggest solutions in the future.

Broader impact. WMM may have benefits in robotics, mixed reality, animation, games, and simulation. We do not see major direct negative ethical or societal consequences of our work.

Acknowledgments and Disclosure of Funding

This project was funded in part by Meta BAIR partners, the BAIR Humanoid Intelligence Center (BAIR HIC), and NSF CAREER (No. 2442491). We used LLMs and agents to write and debug code, help review the literature, and polish the writing.

References

[1] Joao Pedro Araujo, Yanjie Ze, Pei Xu, Jiajun Wu, and C. Karen Liu. Retargeting matters: General motion retargeting for humanoid motion tracking. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2026.

[2] Marianne Arriola, Aaron Gokaslan, Justin Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *International Conference on Learning Representations*, volume 2025, pages 50726–50753, 2025.

[3] Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

[4] Mohammadreza Baharani, Ghazal Alinezhad Noghre, Armin Danesh Pazho, Gabriel Maldonado, and Hamed Tabkhi. Mofm: A large-scale human motion foundation model, 2025. arXiv:2502.05432.

[5] Sherwin Bahmani, Xian Liu, Wang Yifan, Ivan Skorokhodov, Victor Rong, Ziwei Liu, Xihui Liu, Jeong Joon Park, Sergey Tulyakov, Gordon Wetzstein, Andrea Tagliasacchi, and David B. Lindell. Tc4d: Trajectory-conditioned text-to-4d generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[6] Bowen Baker, Ilge Akkaya, Peter Zhokhov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video PreTraining (VPT): Learning to act by watching unlabeled online videos. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[7] German Barquero, Sergio Escalera, and Cristina Palmero. Seamless human motion composition with blended positional encodings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[8] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[9] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

[10] Arunkumar Byravan and Dieter Fox. SE3-Nets: Learning rigid body motion using deep neural networks. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[11] Arunkumar Byravan, Felix Leeb, Franziska Meier, and Dieter Fox. Se3-pose-nets: Structured deep dynamics models for visuomotor planning and control. *arXiv preprint arXiv:1710.00489*, 2017.

[12] Yiyi Cai, Yuhan Wu, Kunhang Li, You Zhou, Bo Zheng, and Haiyang Liu. Flooddiffusion: Tailored diffusion forcing for streaming motion generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2295–2304, 2026.

[13] Angela Castillo, Maria Escobar, Guillaume Jeanneret, Albert Pumarola, Pablo Arbelaez, Ali Thabet, and Artsiom Sanakoyeu. Bodiffusion: Diffusing sparse observations for full-body human motion synthesis, 2023. arXiv:2304.11118.

[14] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. MaskGIT: Masked generative image transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[15] Boyuan Chen, Diego Marti Monso, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[16] Xinzhe Chen, Sihua Ren, Liqi Huang, Haowen Sun, Mingyang Li, Xingyu Chen, Zeyang Liu, and Xuguang Lan. Oasis: Observation-action space alignment via se (3) trajectory prediction for robotic manipulation. *arXiv preprint arXiv:2605.25829*, 2026.

[17] Yi Chen, Yuying Ge, Weiliang Tang, Yizhuo Li, Yixiao Ge, Mingyu Ding, Ying Shan, and Xihui Liu. Moto: Latent motion token as the bridging language for learning robot manipulation from videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[18] Yixiang Chen, Peiyan Li, Yan Huang, Jiabing Yang, Kehan Chen, and Liang Wang. Ec-flow: Enabling versatile robotic manipulation from action-unlabeled videos via embodiment-centric flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[19] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

[20] Xiaoyan Cong, Zekun Li, Zhiyang Dou, Hongyu Li, Omid Taheri, Chuan Guo, Abhay Mittal, Sizhe An, Taku Komura, Wojciech Matusik, Michael J. Black, and Srinath Sridhar. Umo: Unified in-context learning unlocks motion foundation model priors, 2026.

[21] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International Conference on Learning Representations (ICLR)*, 2024.

[22] Chaorui Deng, Deyao Zhu, Kunchang Li, Shi Guang, and Haoqi Fan. Causal diffusion transformers for generative modeling. *arXiv preprint arXiv:2412.12095*, 2024.

[23] Christian Diller and Angela Dai. Cg-hoi: Contact-guided 3d human-object interaction generation. In *CVPR*, 2024.

[24] Yilun Du, Mengjiao Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Joshua B. Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[25] Enes Duran, Nikos Athanasiou, Muhammed Kocabas, Michael J. Black, and Omid Taheri. Fusion: Full-body unified motion prior for body and hands via diffusion, 2026.

[26] Shenyuan Gao, William Liang, Kaiyuan Zheng, Ayaan Malik, Seonghyeon Ye, Sihyun Yu, Wei-Cheng Tseng, Yuzhu Dong, Kaichun Mo, Chen-Hsuan Lin, Qianli Ma, Seungjun Nah, Loic Magne, Jiannan Xiang, Yuqi Xie, Ruijie Zheng, Dantong Niu, You Liang Tan, K. R. Zentner, George Kurian, Suneel Indupuru, Pooya Jannaty, Jinwei Gu, Jun Zhang, Jitendra Malik, Pieter Abbeel, Ming-Yu Liu, Yuke Zhu, Joel Jang, and Linxi Fan. Dreamdojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026.

[27] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4d: Reconstructing and tracking humans with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[28] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 2025.

[29] Nick Hansen, Jyothir SV, Vlad Sobal, Yann LeCun, Xiaolong Wang, and Hao Su. Hierarchical world models as visual whole-body humanoid controllers. In *International Conference on Learning Representations*, volume 2025, pages 62175–62195, 2025.

[30] Zihao He, Bo Ai, Tongzhou Mu, Yulin Liu, Weikang Wan, Jiawei Fu, Yilun Du, Henrik I. Christensen, and Hao Su. Scaling cross-embodiment world models for dexterous manipulation. *arXiv preprint arXiv:2511.01177*, 2025.

[31] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. Short version in NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications.

[32] Sigmund H. Hoeg, Yilun Du, and Olav Egeland. Streaming diffusion policy: Fast policy synthesis with variable noise diffusion models. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2025.

[33] Cheng-Chun Hsu, Bowen Wen, Jie Xu, Yashraj Narang, Xiaolong Wang, Yuke Zhu, Joydeep Biswas, and Stan Birchfield. Spot: Se(3) pose trajectory diffusion for object-centric manipulation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2025.

[34] Haojie Huang, Karl Schmeckpeper, Dian Wang, Ondrej Biza, Yaoyao Qian, Haotian Liu, Mingxi Jia, Robert Platt, and Robin Walters. Imagination policy: Using generative point cloud models for learning manipulation policies. *arXiv preprint arXiv:2406.11740*, 2024.

[35] Junchao Huang, Xinting Hu, Boyao Han, Shaoshuai Shi, Zhuotao Tian, Tianyu He, and Li Jiang. Memory forcing: Spatio-temporal memory for consistent scene generation on minecraft. *arXiv preprint arXiv:2510.03198*, 2025.

[36] Wenlong Huang, Yu-Wei Chao, Arsalan Mousavian, Ming-Yu Liu, Dieter Fox, Kaichun Mo, and Li Fei-Fei. Pointworld: Scaling 3d world models for in-the-wild robotic manipulation, 2026.

[37] Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *Advances in Neural Information Processing Systems*, 38:167283–167308, 2026.

[38] Yanjia Huang, Shuo Liu, Sheng Liu, Qingxiao Xu, Mingyang Wu, Xiangbo Gao, and Zhengzhong Tu. Forge-tree: Diffusion-forcing tree search for long-horizon robot manipulation, 2025. arXiv:2510.21744.

[39] Zixuan Huang, Huaidian Hou, and Dmitry Berenson. Unified multimodal diffusion forcing for forceful manipulation. *arXiv preprint arXiv:2511.04812*, 2025.

[40] Michael Janner, Yilun Du, Joshua B. Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *International Conference on Machine Learning (ICML)*, 2022.

[41] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[42] Chung Min Kim, Brent Yi, Hongsuk Choi, Yi Ma, Ken Goldberg, and Angjoo Kanazawa. PyRoki: A modular toolkit for robot kinematic optimization. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025.

[43] Agelos Kratimenos, Jiahui Lei, and Kostas Daniilidis. DynMF: Neural motion factorization for real-time dynamic view synthesis with 3D gaussian splatting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[44] Yuxuan Kuang, Sungjae Park, Katerina Fragkiadaki, and Shubham Tulsiani. Dex4d: Task-agnostic point track policy for sim-to-real dexterous manipulation, 2026. arXiv:2602.15828.

[45] Kyungmin Lee, Sibeen Kim, Minho Park, Hyunseung Kim, Dongyoon Hwang, Hojoon Lee, and Jaegul Choo. Phuma: Physically-grounded humanoid locomotion dataset. *arXiv preprint arXiv:2510.26236*, 2025.

[46] Seungjae Lee, Yoonkyo Jung, Inkook Chun, Yao-Chih Lee, Zikui Cai, Hongjia Huang, Aayush Talreja, Tan Dat Dao, Yongyuan Liang, Jia-Bin Huang, and Furong Huang. Tracegen: World modeling in 3d trace space enables learning from cross-embodiment videos, 2025.

[47] Jiahui Lei, Yufu Wang, Georgios Pavlakos, Lingjie Liu, and Kostas Daniilidis. GART: Gaussian articulated template models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[48] Jiahui Lei, Kyle Genova, George Kopanas, Noah Snavely, and Leonidas Guibas. Momaps: Semantics-aware scene motion generation with motion maps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[49] Jiahui Lei, Yijia Weng, Adam Harley, Leonidas Guibas, and Kostas Daniilidis. Mosca: Dynamic gaussian fusion from casual videos via 4d motion scaffolds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[50] Jiaman Li, Jiajun Wu, and C. Karen Liu. Object motion guided human motion synthesis. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 2023.

[51] Jiaman Li, Alexander Clegg, Roozbeh Mottaghi, Jiajun Wu, Xavier Puig, and C. Karen Liu. Controllable human-object interaction synthesis. In *European Conference on Computer Vision (ECCV)*, 2024.

[52] Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model. In *Robotics: Science and Systems (RSS)*, 2025.

[53] Zekun Li, Sizhe An, Chengcheng Tang, Chuan Guo, Ivan Shugurov, Linguang Zhang, Amy Zhao, Srinath Sridhar, Lingling Tao, and Abhay Mittal. Llamo: Scaling pretrained language models for unified motion understanding and generation with continuous autoregressive tokens. *arXiv preprint arXiv:2602.12370*, 2026.

[54] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

[55] Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling forcing: Autoregressive long video diffusion in real time, 2025.

[56] Xiaokang Liu, Zechen Bai, Hai Ci, Kevin Yuchen Ma, and Mike Zheng Shou. World-vla-loop: Closed-loop learning of video world model and vla policy. *arXiv preprint arXiv:2602.06508*, 2026.

[57] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations (ICLR)*, 2023.

[58] Yun Liu, Haolin Yang, Xu Si, Ling Liu, Zipeng Li, Yuxiang Zhang, Yebin Liu, and Li Yi. TACO: Benchmarking generalizable bimanual tool-ACtion-object understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[59] Zeyi Liu, Shuang Li, Eric Cousineau, Siyuan Feng, Benjamin Burchfiel, and Shuran Song. Geometry-aware 4d video generation for robot manipulation. *arXiv preprint arXiv:2507.01099*, 2025.

[60] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, 2015.

[61] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11461–11471, 2022.

[62] Corey Lynch, Ayzaan Wahid, Jonathan Tompson, Tianli Ding, James Betker, Robert Baruch, Travis Armstrong, and Pete Florence. Interactive language: Talking to robots in real time. *IEEE Robotics and Automation Letters*, 2023.

[63] Jiangran Lyu, Ziming Li, Xuesong Shi, Chaoyi Xu, Yizhou Wang, and He Wang. Dywa: Dynamics-adaptive world action model for generalizable non-prehensile manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[64] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[65] Vongani H. Maluleke, Kie Horiuchi, Lea Wilken, Evonne Ng, Jitendra Malik, and Angjoo Kanazawa. Diffusion forcing for multi-agent interaction sequence modeling, 2025.

[66] Claire McLean, Makenzie Meendering, Tristan Swartz, Orri Gabbay, Alexandra Olsen, Rachel Jacobs, Nicholas Rosen, Philippe de Bree, Tony Garcia, Gadsden Merrill, Jake Sandakly, Julia Buffalini, Neham Jain, Steven Krenn, Moneish Kumar, Dejan Markovic, Evonne Ng, Fabian Prada, Andrew Saba, Siwei Zhang, Vasu Agrawal, Tim Godisart, Alexander Richard, and Michael Zollhoefer. Embody 3D: A large-scale multimodal motion and behavior dataset. *arXiv preprint arXiv:2510.16258*, 2025.

[67] Yuxuan Mu, Ziyu Zhang, Yi Shi, Dun Yang, Minami Matsumoto, Kotaro Imamura, Guy Tevet, Chuan Guo, Michael Taylor, Chang Shu, et al. Smp: Reusable score-matching motion priors for physics-based character control. *arXiv preprint arXiv:2512.03028*, 2025.

[68] E-Ro Nguyen, Yichi Zhang, Kanchana Ranasinghe, Xiang Li, and Michael S Ryoo. Pixel motion diffusion is what we need for robot control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23663–23672, 2026.

[69] Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *arXiv preprint arXiv:2502.09992*, 2025.

[70] NVIDIA, Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, Daniel Dworakowski, Jiaojiao Fan, Michele Fenzi, Francesco Ferroni, Sanja Fidler, Dieter Fox, Songwei Ge, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025.

[71] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

[72] Chaitanya Patel, Hiroki Nakamura, Yuta Kyuragi, Kazuki Kozuka, Juan Carlos Niebles, and Ehsan Adeli. Uniegomotion: A unified model for egocentric motion reconstruction, forecasting, and generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[73] Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3D with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[74] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[75] Mozhgan Pourkeshavatz, Tianran Liu, and Nicholas Rhinehart. Autoworld: Scaling multi-agent traffic simulation with self-supervised world models. *arXiv preprint arXiv:2603.28963*, 2026.

[76] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[77] Kanchana Ranasinghe, Xiang Li, E-Ro Nguyen, Cristina Mata, Jongwoo Park, and Michael S Ryoo. Pixel motion as universal representation for robot control. *arXiv preprint arXiv:2505.07817*, 2025.

[78] Kanchana Ranasinghe, Honglu Zhou, Yu Fang, Luyu Yang, Le Xue, Ran Xu, Caiming Xiong, Silvio Savarese, Michael S Ryoo, and Juan Carlos Niebles. Future optical flow prediction improves robot control & video generation. *arXiv preprint arXiv:2601.10781*, 2026.

[79] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, Tom Eccles, Jake Bruce, Ali Razavi, Ashley Edwards, Nicolas Heess, Yutian Chen, Raia Hadsell, Oriol Vinyals, Mahyar Bordbar, and Nando de Freitas. A generalist agent. *Transactions on Machine Learning Research*, 2022.

[80] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 36(6):245:1–245:17, 2017.

[81] Yide Shentu, Philipp Wu, Aravind Rajeswaran, and Pieter Abbeel. From LLMs to actions: Latent codes as bridges in hierarchical robot control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024.

[82] Chen Tessler, Yunrong Guo, Ofir Nabati, Gal Chechik, and Xue Bin Peng. Maskedmimic: Unified physics-based character control through masked motion inpainting. *ACM Transactions on Graphics*, 2024.

[83] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022.

[84] Unitree Robotics. Unitree G1 humanoid robot. `https://www.unitree.com/g1`, 2024. Product page.

[85] Qianqian Wang, Vickie Ye, Hang Gao, Weijia Zeng, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4d reconstruction from a single video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9660–9672, 2025.

[86] Xu Wang, Chenkai Xu, Yijie Jin, Jiachun Jin, Hao Zhang, and Zhijie Deng. Diffusion llms can do faster-than-ar inference via discrete diffusion forcing, 2025.

[87] Ye Wang, Sipeng Zheng, Bin Cao, Qianshan Wei, Weishuai Zeng, Qin Jin, and Zongqing Lu. Scaling large motion models with million-level human motions. In *International Conference on Machine Learning (ICML)*, 2025.

[88] Chuan Wen, Xingyu Lin, John So, Kai Chen, Qi Dou, Yang Gao, and Pieter Abbeel. Any-point trajectory modeling for policy learning. In *Robotics: Science and Systems (RSS)*, 2024.

[89] Dejia Xu, Hanwen Liang, Neel P Bhatt, Hezhen Hu, Hanxue Liang, Konstantinos N Plataniotis, and Zhangyang Wang. Comp4d: Llm-guided compositional 4d scene generation. *arXiv preprint arXiv:2403.16993*, 2024.

[90] Sirui Xu, Zhengyuan Li, Yu-Xiong Wang, and Liang-Yan Gui. Interdiff: Generating 3d human-object interactions with physics-informed diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[91] Wenning Xu, Shiyu Fan, Paul Henderson, and Edmond S. L. Ho. Multi-person interaction generation from two-person motion priors. In *ACM SIGGRAPH Conference Papers*, 2025.

[92] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations (ICLR)*, 2025.

[93] Jiacheng Ye, Jiahui Gao, Shansan Gong, Lin Zheng, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Beyond autoregression: Discrete diffusion for complex reasoning and planning. In *International Conference on Learning Representations*, volume 2025, pages 77875–77898, 2025.

[94] Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Sejune Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, Lars Liden, Kimin Lee, Jianfeng Gao, Luke Zettlemoyer, Dieter Fox, and Minjoon Seo. Latent action pretraining from videos. In *International Conference on Learning Representations (ICLR)*, 2025.

[95] Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi Fan, and Joel Jang. World action models are zero-shot policies, 2026.

[96] Chengbo Yuan, Chuan Wen, Tong Zhang, and Yang Gao. General flow as foundation affordance for scalable robot learning. *arXiv preprint arXiv:2401.11439*, 2024.

[97] Yanjie Ze, Zixuan Chen, Joao Pedro Araujo, Zi-ang Cao, Xue Bin Peng, Jiajun Wu, and C. Karen Liu. TWIST: Teleoperated whole-body imitation system. In *Conference on Robot Learning (CoRL)*, 2025.

[98] Jiahao Zhan, Zizhang Li, Hong-Xing Yu, and Jiajun Wu. Perpetualwonder: Long-horizon action-conditioned 4d scene generation, 2026. arXiv:2602.04876.

[99] Mingyuan Zhang, Daisheng Jin, Chenyang Gu, Fangzhou Hong, Zhongang Cai, Jingfang Huang, Chongzhi Zhang, Xinying Guo, Lei Yang, Ying He, and Ziwei Liu. Large motion model for unified multi-modal motion generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[100] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems (RSS)*, 2023.

[101] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: A 3d vision-language-action generative world model. In *International Conference on Machine Learning (ICML)*, 2024.

[102] Artem Zholus, Carl Doersch, et al. Tapnext: Tracking any point (TAP) as next token prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[103] Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning. In *International Conference on Machine Learning (ICML)*, 2025.

[104] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[105] Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. In *Robotics: Science and Systems (RSS)*, 2025.

[106] Silvia Zuffi, Angjoo Kanazawa, David W. Jacobs, and Michael J. Black. 3D menagerie: Modeling the 3D shape and pose of animals. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

A More Ablations

A.1 Rotation scale δ

We vary the rotation scale δ for humanoid retargeting following the setup of Tab. 5. The choice $\delta = 0.2$ reported in Sec. 2.1 of the main paper is empirical and is intended to keep the primal diffusion signal approximately normalized around mean 0 and standard deviation 1. Table 7 compares this choice with $\delta = 0.6$ and $\delta = 1.0$.

Table 7: Rotation-scale ablation on humanoid retargeting.

Setting	SR-seq↑	gerr↓	lerr↓
Paper ($\delta = 0.2$)	0.8690	0.5938	0.0420
$\delta = 0.6$	0.8645	0.6440	0.0449
$\delta = 1.0$	0.8030	0.7709	0.0504

A.2 Semantic and geometric features

We ablate F_{sem} and F_{geo} by setting the corresponding features to zero.

Language Table. Here, F_{sem} consists of three learnable embeddings for object, link-state, and link-action, while F_{geo} is obtained from a PointNet on the local point cloud of the object or link. Removing F_{geo} substantially reduces the success rate (Table 8). The local colored point cloud grounds the language command in specific objects, whereas the semantic feature identifies the entity category rather than which object is which.

Table 8: Feature ablations on Language Table. `noFsem` and `noFgeo` zero the respective features; `noBoth` zeros both.

	Paper	`noFsem`	`noFgeo`	`noBoth`
Avg Succ Rate↑	0.8720	0.8560	0.1960	0.1680

TraceGen. Here, F_{sem} is the DINO feature queried at the pixel location. No F_{geo} is used because the prompt is a point. Removing DINO degrades performance on both EPIC and DROID (Table 9), although the model can still predict future motion from the available geometric information. Semantic features encode "what" the entity is, improving the expressivity of the network.

Table 9: Semantic-feature ablation on TraceGen.

	EPIC			DROID		
Setting	MAE↓	MSE↓	Endpoint↓	MAE↓	MSE↓	Endpoint↓
Paper	2.2564	0.3458	0.6146	1.1718	0.1665	0.2253
`noFsem`	2.4032	0.3627	0.6148	1.2894	0.1876	0.2539

A.3 Consistency with parametric models

For evaluation, we convert predicted poses to each baseline's format: SMPL joint rotations computed from the $SE(3)$ poses, or robot joint angles obtained from relative axis-angles along the kinematic chain with one degree of freedom per joint. Evaluation then follows the corresponding baseline protocol. We measure the consistency between the raw $SE(3)$ predictions and their projected fits in two settings.

SMPL on the OMOMO test set. We compare raw $SE(3)$ joints and bones with the forward-kinematics skeleton of the projected SMPL parameters.

Table 10: Self-consistency of SMPL predictions on the OMOMO test set.

Bone length error	Bone direction error	Raw vs. projected MPJPE
7.21 mm	2.99°	19.15 mm

G1 humanoid test set. We measure the off-hinge-axis rotation and translation removed by projection, together with the difference between raw and forward-kinematics link positions.

Table 11: Self-consistency of predictions on the G1 humanoid test set.

Off-hinge-axis rotation	Off-hinge translation	Raw vs. projected MPJPE
3.57°	13.37 mm	31.38 mm

A.4 Data-generation cost

For parametric and known embodiments, including SMPL bodies, MANO hands, and known robot URDFs, joint or link $SE(3)$ poses can be read directly from saved parameters or actions.

To obtain pose labels for supervision from general RGB videos, we first preprocess the videos to obtain 3D tracks. Table 12 outlines the procedure and reports measurements for a video with 40 frames, $32{,}768$ tracks, and resolution 378×504. The reported total includes the optional full-video object segmentation (VOS) stage, which has not yet been used in this paper.

After the tracking stage, we compute local frames from these tracks. We first define a KNN topology using a topology-aware curve distance: the maximum distance between two trajectories over the times when they are co-visible. Local rigid motion is then obtained by solving a closed-form Procrustes problem.

Table 12: RGB-video-to-3D-track processing cost.

Steps	Time, VRAM
Calibration: resize and run GeoCalib on key frames to undistort intrinsics and estimate the gravity/vertical direction.	4.3 s / 5,856 MB
Geometry: run chunked slice-and-align DA3 (to keep VRAM bounded on long sequences) to estimate depth and camera parameters; also run DA3-metric on key frames to compute the metric scale.	34.6 s / 14,137 MB
Tracking: run 2D tracking using TAPNext and multiple rounds of motion-aware resampling (to keep VRAM bounded and focus more on moving areas), then lift the 2D tracks to 3D at visible time steps using the camera parameters and depth, with depth-boundary and anti-jitter filtering.	64.6 s / 8,151 MB
[OPT] Full VOS (not used in the paper): our pipeline also runs DEVA in auto-mode to segment and track anything in the full video.	27.2 s / 14,079 MB
Total cost	131 s / 14.1 GB

B Technical appendices and supplementary material

B.1 Related Work

Motion Generation and 4D World Modeling. Generative motion modeling has historically fragmented along actor classes. Body, human-human interaction, hand-object interaction, and scene dynamics each use their own state parameterization and task interface [7, 23, 41, 51, 83, 90]. A more recent line of work on unification scales LM backbones, million-scale corpora, compositional priors, and motion-understanding foundations on human motion alone [4, 53, 67, 72, 87, 91, 99], and three further papers consolidate human-side tasks under a single prior. MAGNet [65] models multi-agent body behavior with diffusion-forced canonical pairwise transforms, UMO [20] casts object-aware

human motion as a unified in-context problem, and FUSION [25] trains an unconditional SMPL-X body and hand prior. All of these inherit a body-parametric state and so remain actor-homogeneous within the human family. A second line writes motion into a shared observation space that crosses morphologies. MoMap [48] diffuses pixel-aligned dense 3D motion maps; PointWorld [36] rolls point cloud and gripper flow as a forward simulator; TraceGen [46] flow-matches camera-frame 3D traces conditioned on a single RGB-D start frame and language; GeneralFlow [96] predicts dense 3D flow from RGB-D; EC-Flow [18] maps embodiment-centric pixel flow to end-effector $\mathrm{SE}(3)$ via inverse kinematics; ATM [88] pretrains 2D point tracks as a representation for behavior cloning; TAPNext [102] casts point tracking itself as causal next-token prediction; and sparse-track or sparse-observation diffusion also extends to dexterous manipulation and SMPL completion [13, 44]. These methods share an observation space across actors, but the underlying tokens are translation-only points or dense flow that carry no per-entity identity and discard the rigid-body rotation that drives contact and articulation. A third line generates ambient 4D scene dynamics with explicit trajectories: Comp4D [89] factorizes a 4D world into per-object trajectories plus per-object deformation, and TC4D [5] conditions 4D generation on explicit $\mathrm{SE}(3)$ trajectories, but both target per-scene synthesis rather than a generative prior over heterogeneous moving entities. A piecewise-rigid lineage models scenes directly as per-part $\mathrm{SE}(3)$ transforms: SE3-Nets [10] and SE3-Pose-Nets [11] warp tabletop point clouds through K learned rigid motions conditioned on one robot's actions, OASIS [16] predicts the end-effector pose trajectory for manipulation, and SPOT [33] denoises source-target object pose trajectories from a few demonstrations, yet each remains a single-agent system over a fixed rigid-object set tied to one manipuland or one robot. The closest points of comparison to WMMs are therefore MAGNet and UMO (unified conditioning, human-only), TraceGen and MoMap (trajectory-space world modeling without rotation or per-entity identity), and SPOT together with SE3-Nets ($\mathrm{SE}(3)$-trajectory denoising on a fixed manipuland set), each of which stays inside one of these regimes. WMMs instead encode every moving unit, whether a robot link, hand, body, object, or scene primitive, as a sparse $\mathrm{SE}(3)$ token in a shared rot6d+translation space (Sec. 2.1), so one denoiser is promptable across actor classes and queries arbitrary entities at arbitrary time steps, rather than tracking a fixed dense grid of points or a single morphology family.

World Models and Robot Policy. A unified motion substrate must also support rollout and policy. Pixel-space world-action models keep the substrate in 2D video: DreamDojo [26], DreamZero [95], and Cosmos [70] synthesize frames or extract actions from pretrained video diffusion backbones, and World-VLA-Loop [56] closes the action-perception loop on this pixel substrate with a vision-language-action policy. A related dense-observation model, DyWA [63], runs dynamics-adaptive world-action coupling over single-view point clouds for non-prehensile manipulation, swapping pixel tokens for a 3D point cloud but keeping the world model in a dense observation space. Latent-dynamics planners abstract observations into a learned latent representation: Hierarchical Humanoid WM [29] and DreamerV3 [28] couple latent rollouts with model-predictive control or value-based control, but each instance hard-codes the morphology or the target RL environment. Two recent approaches directly model heterogeneous forward dynamics: Cross-Embodiment WM [30] runs particle-graph rollouts and Geom4D-Robot [59] jointly denoises RGB and pointmap, yet both recover $\mathrm{SE}(3)$ only post-hoc through an additional pose-extraction step rather than treating $\mathrm{SE}(3)$ as the modeling primitive; imagination-policy goal generators [34] and long-horizon scene or traffic generators [75, 98] likewise stay morphology- or domain-locked. Vision-language-action policies form a complementary set of output interfaces over various tokenizations of action [8, 79, 101], and Moto [17] bridges video pretraining to action through a latent motion-token language. Most closely related to our unification approach, UWM [105] and UVA [52] jointly denoise video and action and pursue multi-axis flexibility through shared backbones, but both still tie the world-model side to pixel tokens. WMMs close this gap by predicting $\mathrm{SE}(3)$ trajectories of every tracked entity under a per-token conditioning mask, so one flow-matching model with per-token noise serves as a policy, predictor, and world simulator across arm, humanoid, dexterous hand, human, and tabletop settings, with no morphology-specific head and no separate world and policy modules (Sec. 2.3).

Sequence Modeling with Flexible Conditioning. Diffusion forcing [15] assigns each token an independent noise level, so any subset of tokens can be revealed, in-filled, or rolled out from one trained network, and CausalFusion [22] reconciles autoregressive and diffusion training under the same per-token schedule. This primitive has propagated along several tracks: streaming video adopts diffusion forcing for long-horizon rollouts and live generation [35, 37, 55]; motion and robot control adopt the same schedule on body, action, or interaction tokens [12, 32, 38, 39]; discrete

and masked sequence models reach the same flexible-conditioning behavior through token-level masking schedules [2, 14, 86]; a separate trajectory-level denoising line conditions whole state-action plans on goals or rewards [19, 40]; and earlier flexible-conditioning recipes provide classifier-free guidance, masked image inpainting, and masked motion completion [31, 61, 82]. Each of these instances, however, binds the per-token schedule to a single domain: video frames, one actor's skeletal tokens, one morphology's action horizon, or text. WMMs apply the same per-token noise schedule to $\mathrm{SE}(3)$ trajectories of heterogeneous entities (Sec. 2.2) and extend the trajectory with a context token prefix on a separate timeline for non-causal anchors such as goal states or retargeting exemplars (Sec. 2.1), so one trained network selects between motion synthesis, future prediction, policy rollout, and planning based solely on which tokens are noised at inference (Sec. 2.3).

B.2 Dense generation beyond the trained budget

WMMs are trained on a bounded number of entities and time steps, yet many uses require more entities or time steps than fit in one forward pass (even though RoPE helps to generalize beyond the context window): long-horizon rollouts, dense scene flow, and multi-agent simulations. Splitting into independent chunks fails because two chunks denoised independently disagree at their interface. The *anchored sliding-window* sampler in Alg. 1 instead exploits asynchronous-noise training to keep partially denoised tokens of one chunk as fixed anchors when denoising the next. With a context size S and an anchor count $A < S$, the sampler descends a discrete schedule $\lambda_K > \cdots > \lambda_0 = 0$, so that every forward pass sees S entities at consistent asynchronous levels, exactly the regime used in training with (7). The A shared anchors stitch successive batches into a globally consistent rollout. Asynchronous training plus a non-grid token interface together unlock arbitrarily many entities and time steps at inference.

Algorithm 1 Anchored sliding-window dense sampler.

Require: $S,\ A<S,\ \lambda_K>\cdots>\lambda_0=0,\ \mathcal{E}$ at λ_K
for $k = K, \ldots, 1$ **do**
  $\mathcal{D} \leftarrow \emptyset$
  pick S at λ_k; advance to λ_{k-1}; add to $\mathcal{D}$
  while any entity at λ_k **do**
    A anchors $\sim \mathcal{D}$; $S-A$ fresh at λ_k
    advance fresh to λ_{k-1}; add to $\mathcal{D}$
  end while
end for
return $\mathcal{E}$ at $\lambda_0=0$

B.3 MPC details

At each ACT replanning step, the MPC mode will (i) run one text-conditioned plain inference pass to obtain an imagined goal trace; (ii) build $K{=}4$ candidate action chunks of length 10 (one baseline plus three perturbed by Gaussian noise of $\sigma{=}0.02$ m on the xy components only, with z and rotation frozen); (iii) run the diffusion agent for 12 inner SGD iterations on the candidate xy components with all 10 DDIM denoising steps differentiable and with the language conditioning removed during the optimization rollouts, scoring each candidate by the mean ℓ_2 distance of object positions at the end of the horizon from the imagined goal; and (iv) execute the best candidate for `act_horizon`$=10$ environment steps before replanning. The environment, environment seed, 200-step horizon, and 50 rollouts per family match the plain policy protocol.

B.4 Ablation configurations

Language Table. *no AdaLN* replaces the per-DiT-block shift/scale/gate modulation with a single zero-init linear input fusion at the token-grid input, forcing conditioning to propagate through residuals only. *no per-token noise* replaces the asynchronous per-token noise schedule of Sec. 2.2 with a single shared diffusion timestep t per sample, broadcast to all valid positions; at inference, known tokens are clamped via a noised q-sample with soft blending at strength 0.5.

TraceGen. The reported model for TraceGen uses the ‘tpsqspt‘ layer configuration as in Fig. 3; *no register pool* collapses the middle three layers to plain DiT at the wide dimension, removing the register-compression path and yielding the ‘tpaaapt‘ layer configuration; *plain DiT* collapses all seven layers to plain DiT, removing every architectural primitive and yielding the ‘aaaaaaa‘ layer configuration.

B.5 More Details

A typical WMM transformer architecture uses 8 heads with an embedding dimension of 512 and an FFN comprising a 2-layer MLP with hidden dimension 2048. The attention types are applied sequentially, one type per block. The model is trained on each application dataset using AdamW, a learning rate of 5e-5, a weight decay factor of 1e-3, and a decay step at 300K. For each setting, we train one set of weights on a single A6000 48GB GPU using the largest batch size that fits in memory: Language Table 64; TraceGen 4/12 for P=400/128; OMOMO 16; humanoid 8.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Main claims are verified by experiments.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Is discussed in conclusion section.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: This paper is not a theory paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The main results follows the exsisting benchmark setup and the code and checkpoints will be released.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code will be released at the time when the paper is publisehd, the data are from public accessible source.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The details can be found in the planned code release and the setup is following the exssiting benchmark.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [N/A]

   Justification: The benchmark baselines are not reporting the error bars, the evaluation follows the baselines.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The details can be found in the planned code release and the maximum compute requirement is very low, academic lab with a single 48GB gpu can reproduce.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Fully obey the NeurIPS code of ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This paper studies modeling the motion of the 3D world, has no direct major societal impact. And no harmful potential is found.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: This paper results have no high risk for misues.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [N/A]

Justification: no existing assets to produce the paper.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: the paper does not release new assets.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: the paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: the paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs and Agents are used to improve the writting and help debugging the code.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.