arXiv:2604.27448v1 [cs.CV] 30 Apr 2026

LA-Pose: Latent Action Pretraining Meets Pose Estimation

Zhengqing Wang[1,2*] Saurabh Nair[1*] Prajwal Chidananda[1*]
Pujith Kachana[1] Samuel Li[1] Matthew Brown[1] Yasutaka Furukawa[1,2]

[1]Wayve [2]Simon Fraser University

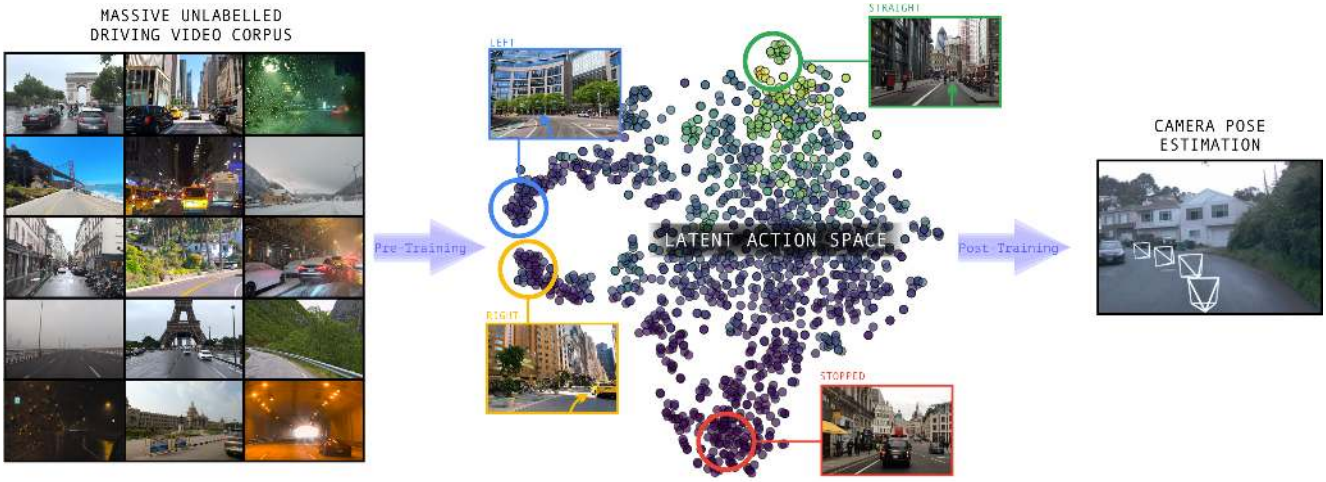

Figure 1. **Overview of LA-Pose.** We introduce a two-stage framework that unifies large-scale *latent action pretraining* with *camera pose estimation*. From millions of unlabeled driving videos, an inverse–forward dynamics model learns *latent actions* that encode inter-frame motion in a fully self-supervised manner. When visualized in T-SNE space, these latent actions exhibit structured clusters that align closely with true ego-motion distributions. We then re-purpose these representations through lightweight supervised post-training on limited 3D-annotated data, enabling feed-forward pose prediction that is both accurate and highly generalizable. **LA-Pose** achieves state-of-the-art pose estimation while requiring *orders of magnitude fewer labeled samples*, demonstrating the power of self-supervised latent action learning for scalable 3D perception.

Abstract

This paper revisits camera pose estimation through the lens of self-supervised pretraining, focusing on inverse-dynamics pretraining as a scalable alternative to the current trend of fully supervised training with 3D annotations. Concretely, we employ inverse- and forward-dynamics models to learn latent action representations, similar to Genie [5] from large-scale driving videos. Our idea is simple yet effective. Existing methods use latent actions in their original capacity, that is, as "action" conditioning of world-models or as proxies of robot "action" parameters in policy networks. Our method, dubbed LA-Pose, repurposes the latent action features as inputs to a camera pose estimator, finetuned on a limited set of high-quality 3D annotations. This formulation enables accurate and generalizable pose prediction while maintaining feed-forward efficiency. Extensive experiments on driving benchmarks show that LA-Pose achieves competitive and even superior performance to state-of-the-art methods while using orders of magnitude less labeled data. Concretely, on the Waymo and PandaSet benchmarks, LA-Pose achieves over 10% higher pose accuracy than recent feed-forward methods. To our knowledge, this work is the first to demonstrate the power of inverse-dynamics self-supervised learning for pose estimation. See the project page for demos and more results: la-pose.github.io.

1. Introduction

Internet-scale pretraining has transformed the landscape of AI, notably through Generative Pretrained Transformers (GPTs) spanning text, image, and video domains [1–3, 7, 8].

*Equal contribution.

Within embodied AI, self-driving stands out as a domain uniquely positioned to ride this wave, fueled by massive collections of driving videos from autonomous fleets and consumer vehicles equipped with dash cameras. For vehicles, motion is a direct consequence of action. Harnessing internet-scale video corpora for vehicle pose estimation—effectively their underlying actions—would be a key technology towards the next generation of self-driving systems.

Concurrently, feed-forward 3D reconstruction techniques such as DUSt3R [34], VGGT [31], and Rig3R [21] are rapidly advancing, achieving impressive accuracy by directly predicting structure and camera poses in a single forward pass. Their success, however, relies on 3D annotations derived from Structure-from-Motion, LiDAR, or simulation engines. High-quality labels are available only in curated datasets that require costly hardware and meticulous calibration, and remain small compared to the vast amount of unlabeled driving video available online. As supervised data has become the bottleneck, surprisingly little effort has been devoted to exploring the potential of self-supervised pretraining—the paradigm that has driven massive breakthroughs in the other domains—for geometry perception tasks such as camera pose estimation studied in this paper.

Our method, dubbed LA-Pose, employs a Genie-style architecture in which an inverse-dynamics module learns latent action representations and a forward-dynamics module uses a latent action to predict the next frame. The latent action model is trained in a fully self-supervised manner, offering strong potential for internet-scale pretraining. Unlike existing methods which use learned latent action representations in their original capacity, for example, as an "action" conditioning of world-models in interactive games [5] or as proxies of robot "action" parameters in policy networks [39], we focus on leveraging latent actions for understanding ego-motion. In self-driving, actions directly manifest as vehicle motion. Consequently, latent actions that model the transition between consecutive frames inherently encode motion change, capturing a compressed representation of pose. LA-Pose repurposes latent actions as inputs to a lightweight pose-estimation head post-trained on a limited set of high-quality 3D annotations. This two-stage design unifies large-scale self-supervised video learning with efficient feed-forward pose prediction.

We evaluate our approach on the Waymo [27] and PandaSet [36] driving benchmarks, where it outperforms state-of-the-art feed-forward 3D reconstruction methods. Extensive quantitative and qualitative analyses demonstrate that our method achieves the highest pose accuracy while requiring significantly less ground-truth 3D annotation. Concretely, on both benchmarks, LA-Pose achieves over 10% higher pose accuracy than recent feed-forward approaches. To our knowledge, this work is the first to demonstrate the power of inverse-dynamics self-supervised learning for pose estimation. We hope that the paper encourages more research in self-supervised learning of geometry perception tasks, towards Internet level scaling.

2. Related Work

Camera Pose Estimation. Classical SfM/VO systems (e.g., COLMAP [26]) remain the default way to obtain camera trajectories for internet imagery. Initial learning-based approaches attempt to directly regress pose using techniques such as convolutions and diffusion [20, 30, 41]. Recent learning-based methods such as DUSt3R [34] directly output aligned dense 3D pointmaps across images, enabling the recovery of relative camera poses. Subsequent works [19, 31, 32, 35, 38] remove DUSt3R's post-inference optimizations and use a feed forward method to directly regress multiple camera poses, actively pushing the state-of-the-art performance. Concurrently, Rig3R [21] uses a dense supervision by predicting pose raymaps to recover the camera poses. These approaches achieve strong results, but are typically bound by their training distribution of labeled 3D data, thus inheriting both their costs and biases. Instead, LA-Pose targets the same feed-forward test-time simplicity while efficiently leveraging large-scale, unlabeled data specifically for pose estimation.

Self Supervised Learning. Recent advances in self-supervised learning have enabled rich, transferable representations across modalities: in images, the seminal DINO [7] framework showed that Vision Transformers [11] can learn semantic features via self-distillation without labels; in videos, the Video-MAE[29], and V-JEPA [3, 4] approaches extended that to spatiotemporal sequences by predicting masked tokens and capturing dynamics; in language, large-scale autoregressive models like GPT demonstrated emergent reasoning capabilities through unsupervised pretraining on massive image-video corpora [1]. More recently, the Scaling 4D Representations [8] explored how masked-autoencoding in large-scale video transformers (up to 22B parameters) improves performance on non-semantic 3D + temporal ("4D") tasks such as depth estimation and camera pose. Some works have similarly explored self-supervised methods for pose understanding [13, 18, 22]; however, these approaches do not leverage large-scale data for pretraining and primarily emphasize photometric reconstruction accuracy, as they are generally designed for novel-view synthesis tasks rather than accurate camera pose estimation. Building on this trajectory, our work brings the predictive self-supervision paradigm into camera pose estimation, proposing a self-supervised pretraining stage that learns geometry-aware representations without explicit 3D labels and thereby bridges the gap between temporal predictive modeling to accurate ego pose estimation.

Latent Action Representation. Latent action representation has recently emerged as a powerful self-supervised learning paradigm for modeling controllable dynamics. Genie-1 [5] introduced the idea of inferring latent "actions" that transform past frames into future ones, learning a compact discrete codebook through an inverse-dynamics bottleneck. The resulting latent actions enable controllable video generation, revealing how temporal transitions can be factorized from visual appearance. Follow-up frameworks extend this idea to robotics, treating latent-action learning as a self-supervised pretraining stage from unlabeled videos, followed by fine-tuning with action-labeled data for output-action prediction and control [10, 15, 25, 28, 39]. Our work explores this concept from a different angle—leveraging latent action representations not for generation or reinforcement learning, but as a way to structure self-supervised learning of ego-motion.

World Models for Video Prediction. World models aim to predict future observations by modeling the dynamics of the environment. In video prediction, a world model learns the temporal evolution of visual scenes, often conditioned on actions that describe how the state changes over time [16, 17, 23, 24]. Classical formulations rely on explicit action labels, which are available in controlled simulation or robotics settings but not in large-scale video data, limiting their scalability to Internet-scale corpora. Recent work bridges this gap by pretraining large world models on unlabeled videos to capture general scene dynamics, and then fine-tuning them with a small amount of action-labeled data to recover explicit control capability [12, 14, 40]. However, such approaches still depend on ground-truth actions for adaptation. Genie overcomes this limitation by introducing latent actions—discrete representations inferred through inverse dynamics—that replace explicit controls as conditioning signals for the forward predictor. This formulation allows a world model to learn structured visual dynamics directly from raw videos without requiring action supervision, enabling scalable training on Internet-scale video collections.

3. LA-Pose

LA-Pose has two stages of training: latent action pretraining and camera pose post-training, as shown in Figure 2. Latent action pretraining is built upon the Genie architecture [5] with several simplifications tailored to our goal of pose estimation. Camera pose post-training learns a lightweight pose estimation head that takes the latent actions from the pretrained inverse dynamics model and estimates relative camera poses and metric scale.

3.1. Latent Action Pretraining

The latent action model of Genie [5] includes an inverse dynamics model and a forward dynamics model, which we adopt to learn latent actions for pose estimation. Genie also has a video tokenizer and an additional forward dynamics model. However, these two components are for high-quality video generation and are not used in our work. This section provides high level description of the Genie architecture and focuses on our modifications.

Image Tokenizer. An input to the system is a sequence of T(=16) frames $(X_1, X_2, \dots X_T)$. The image resolution is 960×448 during training (See section 4 for inference and more details). Each image X_t is tokenized into a set of visual tokens, collectively denoted as s_t. Thus, the sequence of image states is represented as $\{s_1, s_2, \dots, s_T\}$. We follow the standard Vision Transformer design, where patch tokens are added with learnable positional embeddings to preserve spatial layout. To encode temporal information, we use sinusoidal frequency-based temporal embeddings, which are projected through an MLP to match the token dimension. This design allows the model to handle variable frame rates and temporal gaps naturally. The tokens are processed by a 12-layer transformer encoder, resulting in a 15×7×1536 tensor per frame, where 1536 is the feature dimension.

Inverse Dynamics Model. We build on the inverse dynamics model from the Genie architecture, where an ST-Transformer encoder with causal temporal masking produces a sequence of latent actions with one modification. We introduce a 1536-dimensional learnable query token, repeated at all the frames. The query tokens form a set $\{\mathbf{q}_1, ..., \mathbf{q}_{T-1}\}$, which become latent action tokens $\{\mathbf{a}_1, \mathbf{a}_2, \dots \mathbf{a}_{T-1}\}$ at the output of this component. Each query token aggregates information between consecutive frames, serving as the latent action proxy. The causal masking applies to the query tokens in a slightly different way: a_t interacts with frames up to $t+1$. This module handles the tensor dimension as (16×15×7 + 15)×1536. 16 is the number of frames and '+15' denotes the number of query tokens. We further include a pair of three-layer MLPs that compress and de-compress the latent action dimension from 1536 to 50 back to 1536, producing a "compressed version" at the bottleneck. We will evaluate the effects of the compression in Table 2, while the uncompressed version before the MLPs remains our default for the pose estimation stage.

Forward Dynamics Model. Our forward dynamics model also borrows Genie's ST-Transformer for predicting future frames with two modifications. First, we replace the final MLP head with a lightweight four transformer blocks (each with 4 layers of self-attention and feed-forward) that operate on the decoder states and project to the outputs. Second, we use a pretrained VQ-VAE codebook as the prediction target. Specifically, the ground-truth future frames are first encoded by a frozen VQ-VAE encoder into discrete latent codes. Our model predicts the logits over the same codebook for the next frame.

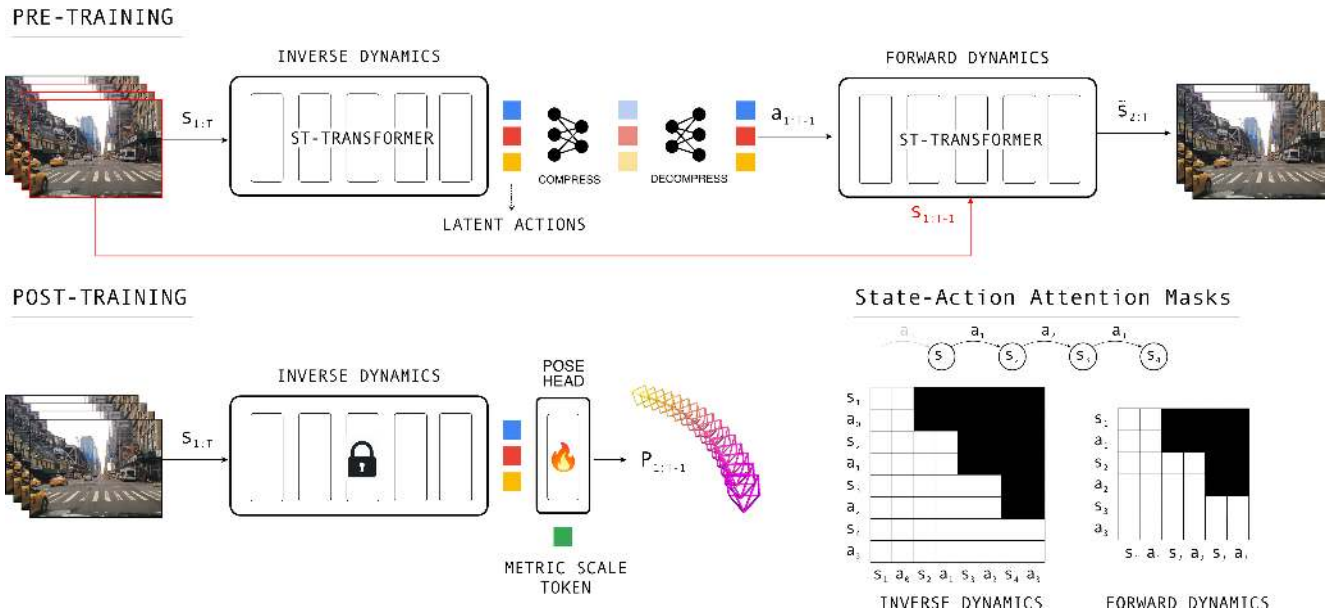

Figure 2. Our framework consists of two stages: latent action pretraining and camera pose post-training. In the pretraining stage (top), an inverse–forward dynamics model learns latent actions from consecutive video frames by predicting future tokens through a self-supervised inverse-dynamics objective. These latent actions encode compact, motion-centric representations of frame-to-frame dynamics. In the post-training stage (bottom left), we attach a lightweight pose estimation head to the pretrained inverse dynamics encoder. The head predicts relative camera translation, rotation (quaternion), field-of-view, and metric scale from the latent actions.

3.2. Camera Pose Post-training

Latent action pretraining captures motion changes and supports effective estimation of camera parameters and metric scale. To facilitate this transfer, we discard the forward dynamics model and attach a relative pose estimation head to the inverse dynamics model. The inverse dynamics model is either frozen or fine-tuned. A small set of training samples with high-quality ground-truth camera poses are used in this post-training step.

Relative Pose Representation. Latent action encodes relative motions from the past frames to the current. Disentangling a metric scale factor from a scale-agnostic representation has been shown to be effective [19, 33]. Therefore, the pose head outputs a metric scale factor through an additional token, alongside poses with scale-agnostic translation components. Specifically, given ground-truth metric motion parameters, we first convert to metric relative motions, where $\{\mathbf{t}_1, \ldots, \mathbf{t}_T\}$ denote per-frame translation components. We compute the average translation magnitude as the metric scale: $s = \text{mean}_i(\|\mathbf{t}_i\|_2)$. We obtain scale-agnostic relative motions by normalizing the translations $\tilde{\mathbf{t}}_i = \mathbf{t}_i / \max(s, \epsilon)$. ϵ is set to 1.0 for numerical stability. Note that we vary the frame rate during training (section 4) and perform this normalization per given frame rate.

Pose Estimation Head. The pose head introduces a single learnable metric scale token ($\mathbb{R}^{1536}$). Together with the latent action tokens $\mathbb{R}^{15\times1536}$, a transformer with self-attention (non-causal) processes all tokens, allowing the metric scale token to aggregate information across the sequence. The decoder outputs are passed through separate MLP heads: one predicts the 7D relative pose (3D translation, 4D quaternion rotation) with the 1D field-of-view, and another predicts a scalar metric scale with exponential activation to ensure positivity and training stability.

3.3. Training

Training Losses. The pretraining loss is a cross-entropy loss between the predicted logits and the ground-truth code indices. The post-training loss is L1 losses on normalized translation, quaternion rotation, field-of-view, and log-space metric scale, where we either freeze or fine-tune the inverse dynamics component.

Training Data. We pre-train on an internal corpus of 10.2 million unlabeled driving video-snippets collected from diverse online and proprietary sources, covering a wide range of environments, traffic densities, and weather conditions. This large-scale dataset provides motion supervision for the inverse-dynamics objective.

For post-training, we use a tiny fraction of that scale, training only on a small set of high-quality labeled data from Waymo [27], nuScenes [6], and Argoverse [9], where accurate LiDAR-calibrated poses provide metric-scale supervision. The Waymo, nuScenes, and Argoverse datasets contain 750, 850, and 700 scenes, respectively, each lasting

approximately 20, 20, and 10 seconds. We generate each training sample by selecting a scene and a frame, while randomizing the frame rate between 1fps and 4fps.

A training sample consists of 16 consecutive frames from a single front-facing camera. To capture a wide range of motion patterns, the frame stride is randomly sampled between 1 fps and 4 fps during pre-training. This temporal jitter encourages the model to learn motion dynamics over both short- and long-term time horizons.

Training Details. Pre-training is conducted on 32 H100 GPUs with a global batch size of 64 for 160k steps using a cosine learning rate schedule (peak 1×10^{-4}, end 4.5×10^{-5}, warm-up 1.5k steps). Post-training is performed on 8 H100 GPUs for 100k steps with a total batch size of 128 using a cosine learning rate schedule (peak 1×10^{-4}, decayed to 0 by the end, warm-up 4k steps). Pre-training completes in approximately four days, and post-training in two days. Notably, our computational cost is significantly lower than that of competing methods. For example, VGGT requires 64 A100 GPUs for nine days of training. These training details correspond to the setup where the pretrained backbone is frozen during post-training, which is our default configuration. Details of the fine-tuning setup are provided in subsection 4.3 together with the ablation study comparing the two configurations.

4. Experiments

4.1. Evaluation Setup

Datasets. We evalaute on the Waymo Open [27] and PandaSet [36] datasets. The pre-training stage is performed on an internal driving video corpus, while post-training and evaluation uses Waymo, nuScenes, and Argoverse datasets (See subsection 3.3). Waymo serves as the in-distribution benchmark, where we use the official training and validation split for LA-Pose post-training and evaluation, respectively. PandaSet serves as a zero-shot benchmark for all evaluated methods. Both datasets contain large-scale, multi-camera driving scenes with precise LiDAR-based camera pose annotations. For evaluation, we uniformly sample frames at 2 fps, corresponding to an 8-second duration over 16 frames).

Metrics. We report three standard metrics: (1) AUC@5, the area under the cumulative error curve up to 5°, computed from pairwise relative rotation and translation angle errors between all frame pairs, reflecting overall pose estimation precision; (2) ATE-S, the scale-invariant aligned trajectory error (RMSE), computed after normalizing trajectories to unit average magnitude and performing global SE(3) alignment, measuring trajectory consistency; and (3) ATE-M, the metric-scale ATE without normalization, reported only when baselines provide metric-scale predictions. When computing scale-invariant metrics, near-stationary sequences (average translation <0.1m) are filtered to ensure numerical stability during normalization.

Table 1. Pose estimation results, reporting area under the curve at an error threshold of 5° *(AUC@5)*, the average aligned trajectory error in scale units *(ATE-S RMSE)*, and in meters *(ATE-M RMSE)*. As Rig3R was trained on the complete PandaSet dataset, it is excluded from the PandaSet evaluation.

Method	AUC@5 ↑ %	ATE-S ↓ $\times 10^{-2}$	ATE-M ↓ m
Waymo			
Rig3R [21]	77.9	3.17	-
VGGT [31]	74.8	1.43	-
MapAnything [19]	65.0	3.00	4.74
LA-Pose	91.4	1.20	0.88
PandaSet (unseen)			
VGGT [31]	75.0	0.99	-
MapAnything [19]	62.4	2.75	7.28
LA-Pose	86.3	1.13	0.86

Baselines. We compare with three recent state-of-the-art pose estimation models: Rig3R [21], VGGT [31], and MapAnything [19]. Rig3R is trained on a strictly broader set of driving datasets than LA-Pose. It includes all datasets used in our training (Waymo, nuScenes, and Argoverse) and additionally PandaSet, KITTI, and several other scene datasets with full 3D supervision. Both VGGT and MapAnything are trained on Mapillary for driving scenarios, with VGGT also using Virtual KITTI2, but both models include a wide variety of non-driving scene datasets with dense geometric labels. Overall, all baseline methods rely on substantially larger amounts of supervised 3D data, whereas LA-Pose combines large-scale self-supervised pre-training with fine-tuning on a limited number of labeled driving sequences.

4.2. Main Results

Quantitative Comparison. Table 1 presents quantitative results on the Waymo and PandaSet benchmarks. On Waymo, LA-Pose attains an AUC@5 of 91.4% and an ATE of 1.20×10^{-2}. On the unseen PandaSet benchmark, our model maintains strong generalization with an AUC@5 of 86.3%, surpassing all baselines, while the ATE remains comparable to VGGT (1.13×10^{-2} vs. 0.99×10^{-2}).

Table 1 reports the mean of each metric across all samples. To provide a more detailed view, we analyze the distribution of AUC@5 scores in Figure 4. LA-Pose not only achieves a higher average AUC but also exhibits substantially lower variance across test scenes. Most of our sam-

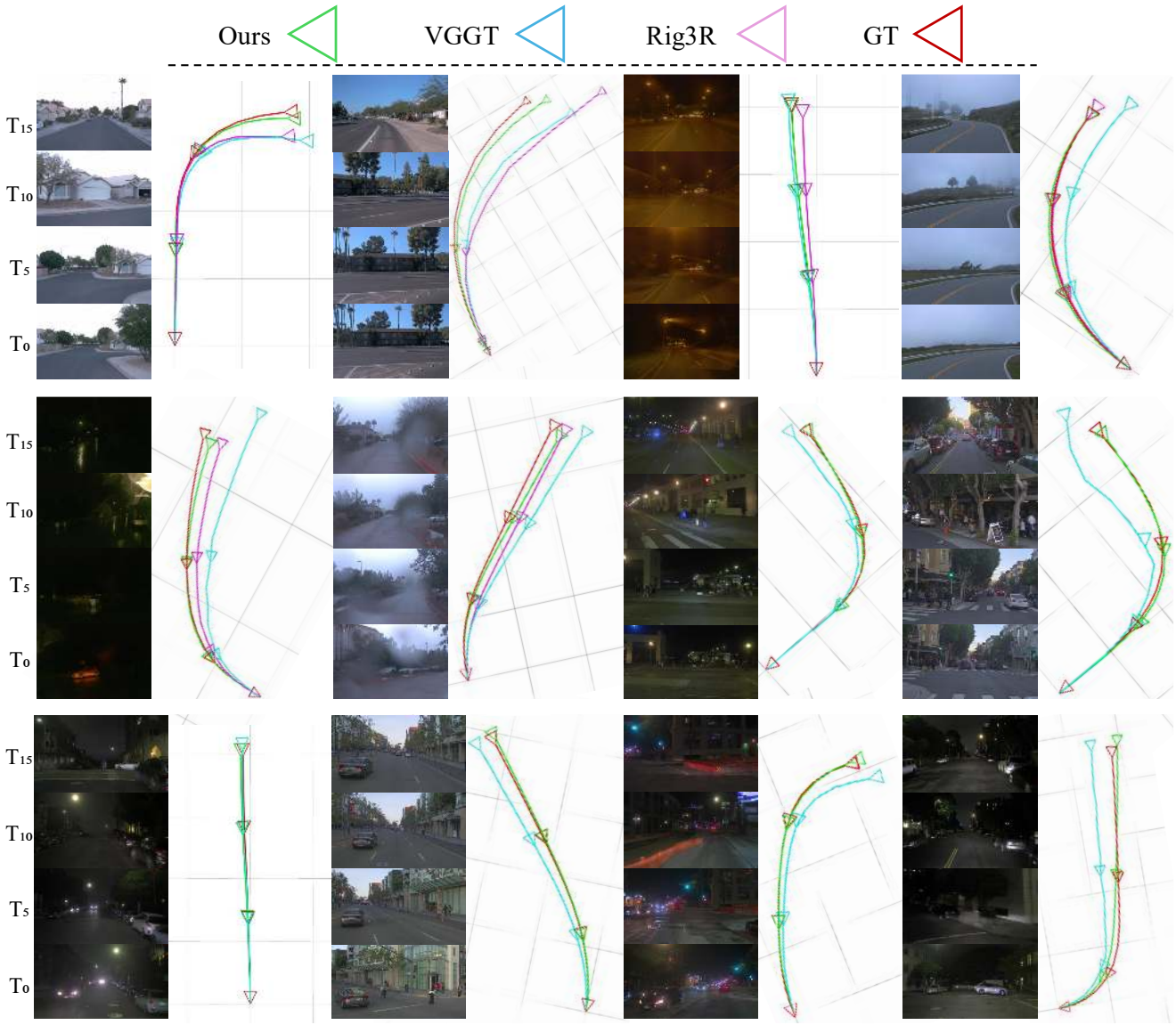

Figure 3. Qualitative results of camera pose estimation. Comparison of predicted camera trajectories: Ours (green), Rig3R [21] (magenta), VGGT [31] (cyan), and ground truth (red). Camera frustums are visualized at timestamps 0, 5, 10, and 15, with trajectory lines connecting all camera positions in the sequence. The first six examples show results on the Waymo dataset, while the last six are from PandaSet. All trajectories are projected onto the xz plane for visualization clarity.

ples cluster near perfect accuracy, while VGGT shows a wider spread with a long tail of low-performing cases. This indicates that LA-Pose delivers more consistent and reliable pose estimation across diverse scenarios, rather than excelling only on easy sequences.

Qualitative Analysis. Figure 3 visualizes representative examples of estimated camera poses and corresponding images. To provide a balanced view beyond overall averages, we refer to the AUC@5 distribution in Figure 4 and focus our qualitative analysis primarily on the more challenging cases for both LA-Pose and VGGT. From the 185 evaluation samples, we examined those with relatively lower AUC scores and selected a diverse subset capturing different conditions, including rain, nighttime, fog, and sharp turns. Even in these difficult scenarios, LA-Pose consistently produces stable and geometrically coherent trajectories, highlighting its robustness under adverse conditions. Notably, the post-training data are dominated by simple straight-motion sequences without targeted sampling of rare cases. This robustness emerges naturally from large-scale self-supervised pretraining, which exposes the model to diverse motion patterns and visual appearances.

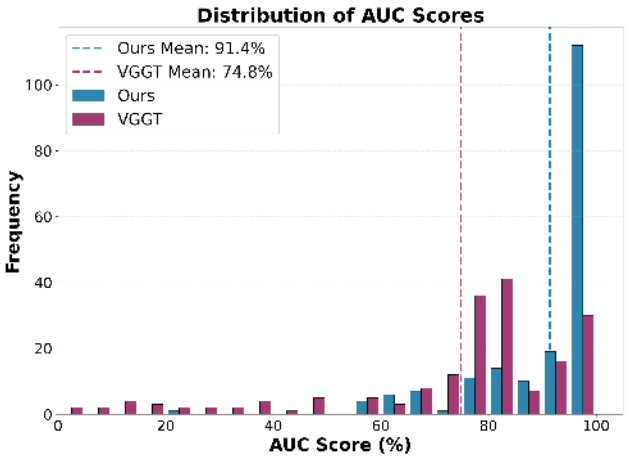

Figure 4. Distribution of pose estimation AUC@5 for LA-Pose and VGGT on the Waymo Open Dataset.

4.3. Ablation on freezing/finetuning the backbone

We study the effect of freezing versus fine-tuning the pre-trained inverse dynamics model. Figure 5 compares the performance on the Waymo and PandaSet benchmarks. The x-axis indicates the number of pre-training samples. Note that a batch size is 64, and 40k post-training steps equal to roughly 2.6 M training samples. 80k steps correspond to 5.2 M. We report the ATE-M metric.

The figure shows that both configurations achieve comparable performance on the in-domain Waymo benchmark. On the zero-shot PandaSet benchmark, the fine-tuning version degrades significantly, indicating that freezing the backbone preserves the motion priors learned during pre-training and achieves superior generalization.

4.4. Ablation on Latent Action Dimension

We study how the dimension of the latent action feature influences both self-supervised pre-training and downstream pose estimation. To save computation, we conduct this ablation using 8 × H100 GPUs with batch size 16 for both pre-training and finetuning. Pre-training was performed over a smaller set 3.2M samples, and post-training was limited to 60k steps.

Table 2 summarizes the effect of varying the latent action dimensionality. A larger latent space (e.g., 1536-D) achieves lower reconstruction loss during pre-training, as it directly encodes dense motion flow and appearance cues, making the forward dynamics prediction easier. However, this leads to information leakage and weaker abstraction of ego-motion, which is detrimental to pose estimation. In contrast, a smaller latent space (e.g., 50-D) yields higher pre-training loss but promotes compact, motion-centric representations that transfer more effectively to downstream pose estimation. When freezing the pretrained inverse dynamics encoder and training only the pose estimation head, the 50-D latent achieves nearly the same AUC@5 as the 1536-D variant while significantly improving ATE-M, confirming that stronger compression enhances motion awareness and metric-scale consistency.

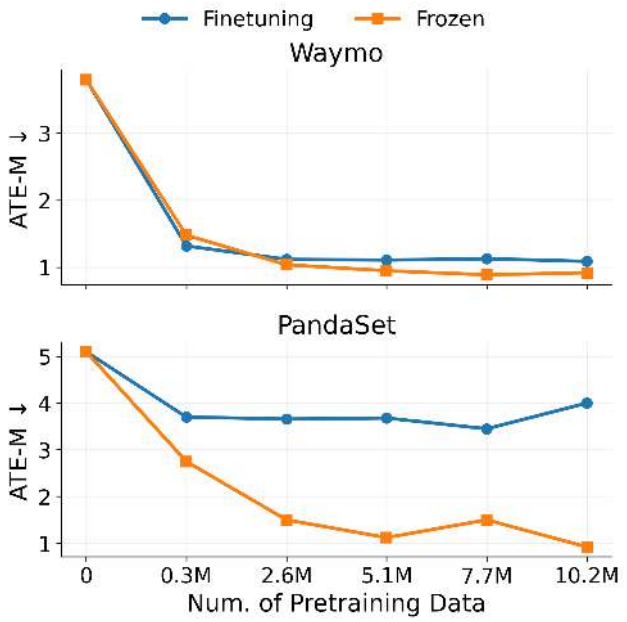

Figure 5. Comparison of pose post-training with frozen and fine-tuned inverse-dynamics encoders. The x-axis shows pretraining data scale. The y-axis reports metric-scale ATE-M (↓). Both settings perform similarly on Waymo, while the frozen backbone generalizes markedly better to the unseen PandaSet.

Table 2. **Ablation on latent action dimension.** Comparison between different latent dimensions under the *post-training* setting on Waymo. Larger latent spaces yield lower reconstruction loss during pre-training but enable information leakage, leading to degraded downstream pose estimation. Pre-training losses at 100k and 200k steps are measured as cross-entropy on the VQ-VAE code prediction.

Latent Dim.	Pre-training Loss		Post-training	
	@100k ↓	@200k ↓	AUC@5 ↑	ATE-M ↓
50	1.87	1.67	85.4	1.62
1536	1.35	1.15	86.5	1.94

4.5. Robustness to Frame Sampling Rate

We evaluate the robustness of our model under varying temporal sampling rates, that is, the frame rate of an input video. A higher fps (smaller temporal gap) provides denser motion cues, while a lower fps corresponds to sparser observations. As shown in Table 3, LA-Pose consistently out-

Table 3. **Robustness to frame sampling rate.** Comparison between LA-Pose and VGGT [31] under different frame rates on the Waymo benchmark. LA-Pose achieves consistently lower trajectory error (ATE-S) and higher pose accuracy (AUC@5).

FPS	Method	AUC@5 ↑ (%)	ATE-S ↓ ($\times 10^{-2}$)
4.0	VGGT	74.1	1.03
	LA-Pose	93.4	0.87
1.3	VGGT	75.0	1.21
	LA-Pose	88.6	1.20
1.0	VGGT	74.6	1.43
	LA-Pose	85.7	1.16

performs VGGT [31] across all fps settings. While both methods experience a mild decline in performance at lower fps, LA-Pose maintains stable accuracy and low translation error, demonstrating strong temporal robustness.

5. Limitations, Future work, and Conclusion

LA-Pose is a self-supervised framework bridging large-scale video pretraining with efficient camera pose estimation. By repurposing latent action features from pretraining as motion-centric features for pose estimation, LA-Pose demonstrates that scalable video pretraining effectively substitutes costly 3D supervision. Experiments on the Waymo and PandaSet benchmarks show that LA-Pose achieves state-of-the-art accuracy with significantly fewer labeled data. Despite its strong performance, we observe degraded accuracy in rare cases such as reverse motion (*i.e.*, backing up), as illustrated in Figure 6. These scenarios are underrepresented in the supervised post-training data, leading to less stable pose estimates. Future work includes scaling up the pretraining dataset to improve robustness in such rare cases. More broadly, our formulation of self-supervised video pretraining followed by supervised pose fine-tuning is applicable to other domains. Going beyond driving and extending pretraining to in-the-wild embodied videos–including casual recordings from diverse agents, environments, and camera configurations–could yield general geometric priors transferable across domains.

Acknowledgments

We thank Jaskaran Singh Sodhi and Saloni Puran Parekh for their help with data construction, and Anner De Jong for engineering support. We are grateful to Thomas Kollar and Gianluca Corrado for reviewing the paper and providing valuable feedback. We also thank Shreyas Rajesh and Soham Phade for their early contributions to the idea of Latent-Actions.

Figure 6. **Failure case under reverse motion.** Performance degrades when the vehicle moves backward, a rare condition in the supervised training set. Despite this distribution gap, the pretrained backbone still produces partially consistent trajectories.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1, 2

[2] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15619–15629, 2023.

[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. 1, 2

[4] Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *arXiv preprint arXiv:2404.08471*, 2024. 2

[5] Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024. 1, 2, 3

[6] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 4

[7] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 1, 2

[8] João Carreira, Dilara Gokay, Michael King, Chuhan Zhang, Ignacio Rocco, Aravindh Mahendran, Thomas Albert Keck, Joseph Heyward, Skanda Koppula, Etienne Pot, et al. Scaling 4d representations. *arXiv preprint arXiv:2412.15212*, 2024. 1, 2

[9] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, et al. Argoverse: 3d tracking and forecasting with rich maps. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8748–8757, 2019. 4

[10] Zichen Cui, Hengkai Pan, Aadhithya Iyer, Siddhant Haldar, and Lerrel Pinto. Dynamo: In-domain dynamics pretraining for visuo-motor control. *Advances in Neural Information Processing Systems*, 37:33933–33961, 2024. 3

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021. 2

[12] Gao et al. Vista: A generalizable driving world model with high fidelity and versatile controllability. *Advances in Neural Information Processing Systems*, 37:91560–91596, 2024. 3

[13] Zhou et al. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[14] Ruili Feng, Han Zhang, Zhantao Yang, Jie Xiao, Zhilei Shu, Zhiheng Liu, Andy Zheng, Yukun Huang, Yu Liu, and Hongyang Zhang. The matrix: Infinite-horizon world generation with real-time moving control. *arXiv preprint arXiv:2412.03568*, 2024. 3

[15] Shenyuan Gao, Siyuan Zhou, Yilun Du, Jun Zhang, and Chuang Gan. Adaworld: Learning adaptable world models with latent actions. *arXiv preprint arXiv:2503.18938*, 2025. 3

[16] Mariam Hassan, Sebastian Stapf, Ahmad Rahimi, Pedro M B Rezende, Yasaman Haghighi, David Brüggemann, Isinsu Katircioglu, Lin Zhang, Xiaoran Chen, Suman Saha, Marco Cannici, Elie Aljalbout, Botao Ye, Xi Wang, Aram Davtyan, Mathieu Salzmann, Davide Scaramuzza, Marc Pollefeys, Paolo Favaro, and Alexandre Alahi. Gem: A generalizable ego-vision multimodal world model for fine-grained ego-motion, object dynamics, and scene composition control, 2024. 3

[17] Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023. 3

[18] Hanwen Jiang, Hao Tan, Peng Wang, Haian Jin, Yue Zhao, Sai Bi, Kai Zhang, Fujun Luan, Kalyan Sunkavalli, Qixing Huang, and Georgios Pavlakos. Rayzer: A self-supervised large view synthesis model. 2025. 2

[19] Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, et al. Mapanything: Universal feed-forward metric 3d reconstruction. *arXiv preprint arXiv:2509.13414*, 2025. 2, 4, 5

[20] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization, 2016. 2

[21] Samuel Li, Pujith Kachana, Prajwal Chidananda, Saurabh Nair, Yasutaka Furukawa, and Matthew Brown. Rig3r: Rig-aware conditioning for learned 3d reconstruction. *arXiv preprint arXiv:2506.02265*, 2025. 2, 5, 6

[22] Thomas W. Mitchel, Hyunwoo Ryu, and Vincent Sitzmann. True self-supervised novel view synthesis is transferable, 2025. 2

[23] 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, Yunhao Ge, Jinwei Gu, Siddharth Gururani, Ethan He, Jiahui Huang, Jacob Huffman, Pooya Jannaty, Jingyi Jin, Seung Wook Kim, Gergely Klár, Grace Lam, Shiyi Lan, Laura Leal-Taixe, Anqi Li, Zhaoshuo Li, Chen-Hsuan Lin, Tsung-Yi Lin, Huan Ling, Ming-Yu Liu, Xian Liu, Alice Luo, Qianli Ma, Hanzi Mao, Kaichun Mo, Arsalan Mousavian, Seungjun Nah, Sriharsha Niverty, David Page, Despoina Paschalidou, Zeeshan Patel, Lindsey Pavao, Morteza Ramezanali, Fitsum Reda, Xiaowei Ren, Vasanth Rao Naik Sabavat, Ed Schmerling, Stella Shi, Bartosz Stefaniak, Shitao Tang, Lyne Tchapmi, Przemek Tredak, Wei-Cheng Tseng, Jibin Varghese, Hao Wang, Haoxiang Wang, Heng Wang, Ting-Chun Wang, Fangyin Wei, Xinyue Wei, Jay Zhangjie Wu, Jiashu Xu, Wei Yang, Lin Yen-Chen, Xiaohui Zeng, Yu Zeng, Jing Zhang, Qinsheng Zhang, Yuxuan Zhang, Qingqing Zhao, and Artur Zolkowski. Cosmos world foundation model platform for physical ai, 2025. 3

[24] Lloyd Russell, Anthony Hu, Lorenzo Bertoni, George Fedoseev, Jamie Shotton, Elahe Arani, and Gianluca Corrado. Gaia-2: A controllable multi-view generative world model for autonomous driving. *arXiv preprint arXiv:2503.20523*, 2025. 3

[25] Dominik Schmidt and Minqi Jiang. Learning to act without actions, 2024. 3

[26] Johannes L. Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[27] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020. 2, 4, 5

[28] Bahey Tharwat, Yara Nasser, Ali Abouzeid, and Ian Reid.

Latent action pretraining through world modeling. *arXiv preprint arXiv:2509.18428*, 2025. 3

[29] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training, 2022. 2

[30] Jianyuan Wang, Christian Rupprecht, and David Novotny. Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment, 2024. 2

[31] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025. 2, 5, 6, 8, 1

[32] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 10510–10522, 2025. 2

[33] Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *arXiv preprint arXiv:2507.02546*, 2025. 4

[34] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024. 2

[35] Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Scalable permutation-equivariant visual geometry learning, 2025. 2

[36] Pengchuan Xiao, Zhenlei Shao, Steven Hao, Zishuo Zhang, Xiaolin Chai, Judy Jiao, Zesong Li, Jian Wu, Kai Sun, Kun Jiang, et al. Pandaset: Advanced sensor suite dataset for autonomous driving. In *2021 IEEE international intelligent transportation systems conference (ITSC)*, pages 3095–3101. IEEE, 2021. 2, 5

[37] Jiazhi Yang, Shenyuan Gao, Yihang Qiu, Li Chen, Tianyu Li, Bo Dai, Kashyap Chitta, Penghao Wu, Jia Zeng, Ping Luo, et al. Genad: Generalized predictive model for autonomous driving. *arXiv preprint arXiv:2403.09630*, 2024. 1

[38] Jianing Yang, Alexander Sax, Kevin J. Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2

[39] Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Sejune Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, et al. Latent action pretraining from videos. *arXiv preprint arXiv:2410.11758*, 2024. 2, 3

[40] Jiwen Yu, Yiran Qin, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Gamefactory: Creating new games with generative interactive videos. *arXiv preprint arXiv:2501.08325*, 2025. 3

[41] Jason Y. Zhang, Amy Lin, Moneish Kumar, Tzu-Hsuan Yang, Deva Ramanan, and Shubham Tulsiani. Cameras as rays: Pose estimation via ray diffusion, 2024. 2

LA-Pose: Latent Action Pretraining Meets Pose Estimation

Supplementary Material

The supplementary document provides additional qualitative visualizations and analysis:

- Qualitative comparison between LA-Pose and VGGT [31] under the low frame rate (1 fps) setting on Waymo. (§6)
- Additional qualitative results on the OpenDV–YouTube dataset. (§7)
- Analysis of failure modes across different motion regimes on the Waymo validation set. (§8)

6. Qualitative Results under Low Frame Rate

Figure 7 presents additional qualitative comparisons between LA-Pose and VGGT [31] under the low frame rate (1 fps) setting on the Waymo dataset. All visualizations follow the same protocol as in the main paper, where predicted camera trajectories are projected to the xz plane with camera frustums shown at frames 0, 5, 10, and 15.

At this extremely sparse temporal sampling, VGGT suffers from noticeable drift and unstable pose transitions, especially along long or turning trajectories. In contrast, LA-Pose produces smoother and more consistent camera trajectories, maintaining stable motion even with large temporal gaps between frames. These qualitative results highlight the robustness of our learned latent action representation when operating under low frame rate conditions.

7. Qualitative Results on OpenDV–YouTube

Figure 8 shows qualitative results of LA-Pose on the OpenDV–YouTube dataset [37]. The OpenDV–YouTube dataset [37] is a large-scale collection of unconstrained driving videos gathered from public YouTube channels. It forms the main component of OpenDV-2K, spanning over 1700 hours of front-view recordings captured across more than 40 countries and 240 cities, far exceeding the geographic coverage of our post-training datasets such as Waymo (San Francisco, Phoenix), nuScenes (Boston and Singapore), and Argoverse (six U.S. cities).This vast diversity covers a wide range of road types, weather conditions, lighting, and camera setups, making OpenDV–YouTube an extremely challenging for evaluating generalization.

OpenDV–YouTube consists of uncalibrated front-view driving videos collected from YouTube, recorded with diverse in-vehicle cameras of unknown intrinsic and extrinsic parameters. Since the dataset contains only raw RGB frames without ground-truth camera poses, we visualize our predicted camera trajectories to qualitatively assess pose consistency and realism.

We attribute this strong generalization capability to our pre-training stage, which learns a robust video representation from large-scale unlabeled data. The frozen backbone, pre-trained on massive driving video corpora, serves as a powerful feature extractor that captures high-level motion patterns and latent action structures. This foundation enables the model to transfer effectively to unseen conditions and datasets like OpenDV–YouTube, maintaining stable pose predictions even in unstructured, out-of-distribution environments.

8. Failure Mode Analysis

To further understand the limitations of our method, we analyze pose estimation performance across different trajectory curvatures and accelerations on the Waymo validation set. We categorize trajectories into bins based on curvature and acceleration to examine model performance under different motion regimes. Curvature is defined as $\kappa = d\psi/ds$, where ψ denotes the vehicle heading and s is the trajectory arc length. We divide curvature into three ranges: small (< 0.01 m^{-1}), medium (0.01–0.1 m^{-1}), and large (> 0.1 m^{-1}). Similarly, acceleration magnitude is divided into three ranges: < 0.3, 0.3–0.8, and > 0.8 m/s^2.

Table 4 reports AUC@5 (%) under these motion regimes. We observe that performance is lower on medium-curvature trajectories compared to straight or sharp-turn motions. Medium-curvature trajectories correspond to gradual steering behaviors where frame-to-frame geometric changes are subtle. In these situations, visual motion cues between consecutive frames are weak, making the motion representation learned through future-frame prediction less discriminative. By contrast, straight trajectories exhibit stable ego-motion patterns, while sharp turns introduce stronger geometric changes that are easier for the model to capture.

Table 4. AUC@5 (%) across different trajectory curvatures and accelerations on the Waymo validation set.

	Small	Medium	Large
Curvature	94.50	78.32	91.22
Acceleration	92.81	90.96	88.25

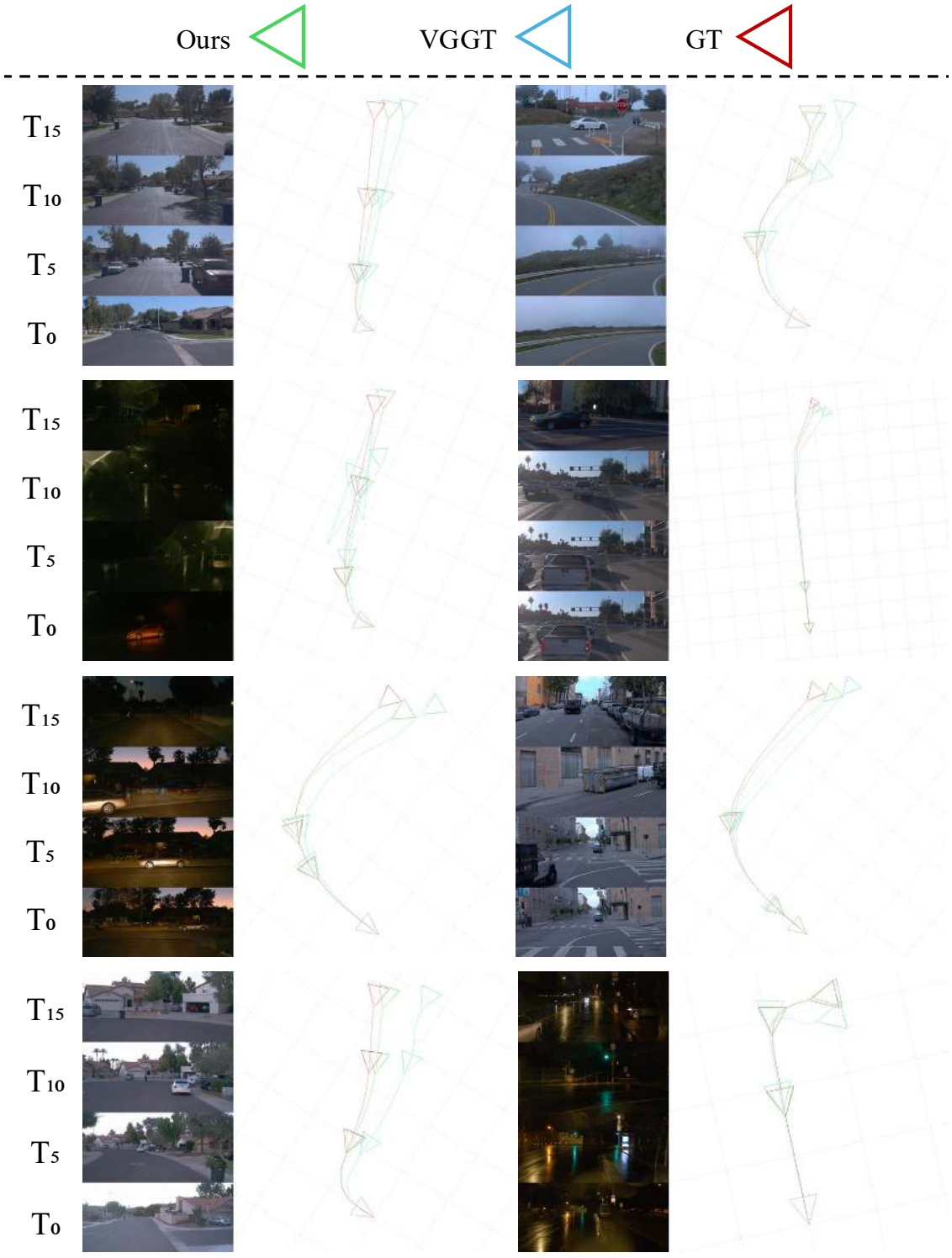

Figure 7. **Qualitative results under low frame rate (1 fps) on Waymo.** Each example shows camera poses projected onto the xz plane, with frustums drawn at frames 0, 5, 10, and 15. LA-Pose (green) maintains stable and temporally consistent motion across the sequence, whereas VGGT [31] (cyan) exhibits noticeable drift and discontinuities under sparse temporal sampling.

Figure 8. **Qualitative results on OpenDV–YouTube.** Each example shows scenes from diverse cities and viewpoints collected from online YouTube driving videos. LA-Pose produces stable and temporally consistent trajectories across a wide variety of conditions, including urban streets, highways, and curved mountain roads. The results qualitatively demonstrate strong generalization from our pre-trained backbone to uncalibrated, in-the-wild videos.