Track3R: Joint Point Map and Trajectory Prior for Spatiotemporal 3D Understanding

Seong Hyeon Park
KAIST
seonghyp@kaist.ac.kr

Jinwoo Shin
KAIST and RLWRLD
jinwoos@kaist.ac.kr

Abstract

Understanding the 3D world from 2D monocular videos is a crucial ability for AI. Recently, to tackle this underdetermined task, end-to-end 3D geometry priors have been sought after, such as pre-trained point map models at scale. These models enable robust 3D understanding from casually taken videos, providing accurate object shapes disentangled from uncertain camera parameters. However, they still struggle when affected by object deformation and dynamics, failing to establish consistent correspondence over the frames. Furthermore, their architectures are typically limited to pairwise frame processing, which is insufficient for capturing complex motion dynamics over extended sequences. To address these limitations, we introduce Track3R, a novel framework that integrates a new architecture and task to jointly predict point map and motion trajectories across multiple frames from video input. Specifically, our key idea is modeling two disentangled trajectories for each point: one representing object motion and the other camera poses. This design not only can enable understanding of the 3D object dynamics, but also facilitates the learning of more robust priors for 3D shapes in dynamic scenes. In our experiments, Track3R demonstrates significant improvements in a joint point mapping and 3D motion estimation task for dynamic scenes, such as 25.8% improvements in the motion estimation, and 15.7% in the point mapping accuracy.

1 Introduction

The recent advance of 3D prior models by the point mapping frameworks has enabled robust and accurate 3D understanding in casually taken video frames. Unlike the 3D reconstruction methods that rely on external matching, depth, and pose estimation priors [1], these models directly learn 3D shape priors via end-to-end designs predicting dense 3D points directly from 2D frames. Notably, DUSt3R [2] has introduced the pair-wise point mapping, which, given a pair of two image frames with unknown camera parameters, maps every pixel in one frame to 3D points in the other frame's view. This design allows robust 3D representation disentangling the 3D shapes and the camera motion, providing a strong prior trained at scale.

However, current methods are often challenged when given videos capturing dynamic scenes, affected by moving and deforming objects. For example, the point map task accounts for only variable camera poses, but cannot establish a consistent motion trajectory of the 3D points over the frames. Furthermore, the typical model architectures suffer from the constrained temporal window size of 2 frames, which hinders modeling complex dynamics spanning over wider windows. Although there have been approaches to mitigate the problem, such as injecting motion estimation priors [3], memory bank architecture [4], they cannot generalize well to various tasks in dynamic scenes and fail to learn holistic priors that can disentangle the object dynamics.

To tackle this problem, we propose a new framework which can jointly predict the point map and their motion trajectory from a multiple number of frames in the end-to-end manner, coined Track3R.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

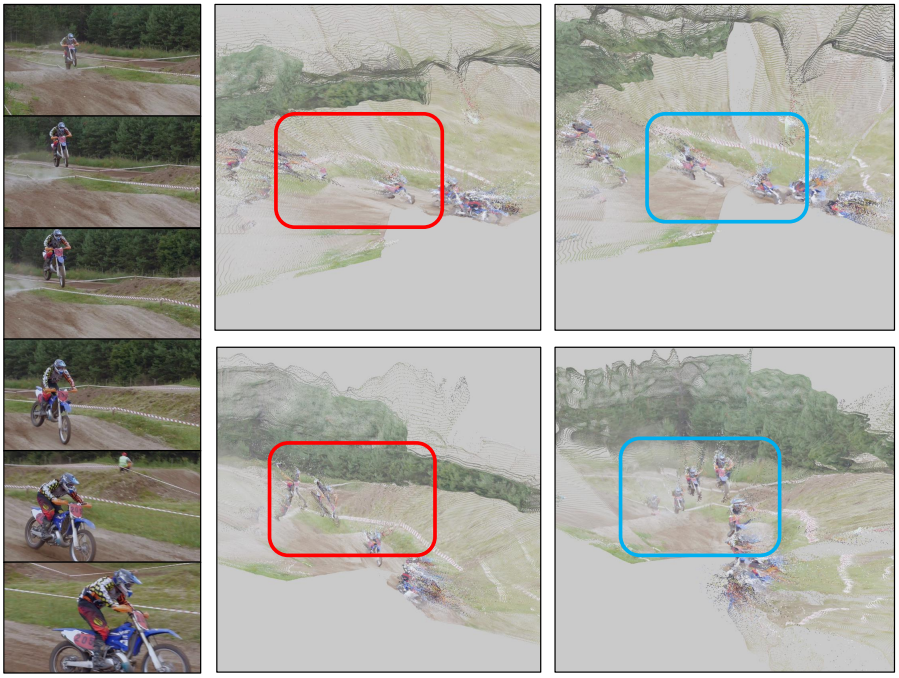

Figure 1: **The joint point mapping and trajectory prediction.** The outputs of a pair-wise point mapping [3] and our method are compared, processing 6 distant frames of a video [5] capturing dynamic scenes. With the first frame as the reference, the 3D points are predicted for other frames.

Specifically, our key idea is modeling two disentangled trajectories for each point: one representing the 3D motion and the other representing camera views. We enable this task via generalizing a well-adopted point mapping framework, the Siamese transformer architecture with two parallel decoders [6, 7] for the pair-wise point mapping, where the first and the second decoders process distinct frames.

To be specific, assuming W input frames, we introduce a factorized temporal attention over the tokens in the same spatial locations of the frames [8], which enables predicting point maps beyond the 2-frame constraint. For instance, while the vanilla architecture would perform $\texttt{Permutation}(W, 2)$ pair-wise iterations, our enhanced architecture would produce an equivalent set of point map sequences in a single forward pass,[1] as depicted in Figure 1. Then, we modify the semantics represented by these sequences: the output by the first decoder represents the 3D motion trajectory in a fixed camera coordinate system, and the other by the second decoder to be 3D points in the variable camera views.

Our model can readily utilize 3D shape priors pre-trained on static scenes at scale, which enables a training focused on fine-tuning the model in dynamic scenes for learning motion priors. For example, we employ 3D trajectory annotations from human motion datasets [9] as the ground truth for training, and synthesize the input frames via a dedicated 3D rendering framework for the human bodies [10]. While the dynamic video data in prior art, such as Kubric [11] and Point Odyssey [12], are with a sparse set of trajectory annotations, human motion datasets provide a denser set of point trajectories, which can facilitate learning stronger priors for joint 3D geometry and motion in dynamic scenes.

Track3R not only can provide more accurate outputs for the tasks involving 3D motion estimation, but also enables the learning of more robust priors for 3D geometry in dynamic scenes. For example, it achieves significantly improved experimental results compared to strong baselines, *e.g.*, relative 25.8% on 3D motion estimation and 15.7% on point mapping tasks.

[1]We note that the vanilla architecture and the enhanced architecture are essentially identical if $W = 2$.

2 Related Work

2.1 3D prior learning

The traditional approaches have focused on learning priors for matching and depth features, which only partially describe the 3D configuration in images and prone to the error accumulation in the independent prior models. Since the introduction of end-to-end architectures, such as the image-based neural fields [13] and the feed-forward Gaussian splatting [14], the direct 3D prior learning from image inputs has gained research interest. However, their main focus is typically on reconstructing photometric colors of the image pixels, assuming the camera parameters should be known a priori, which hinders learning robust 3D representation from casually-taken videos [15].

Recently, the point mapping task [2] has been introduced, which choose to directly regress the 3D point coordinates without requiring known camera configurations. The point map can effectively disentangle the influence of camera motion in the 3D geometry, which has been shown to enable learning robust 3D shape priors trained at scale. Our goal is to further generalizing the point mapping 3D prior learning for dynamic scenes, which we further discuss as follows.

2.2 3D tracking models

Recently, the conventional 2D video point tracking tasks [16] have evolved towards understanding motion in 3D space [17]. Since the goal of the 3D tracking is predicting the motion trajectory and detecting the occlusions of a query point in video frames, assuming that the camera poses are known a priori, the state-of-the-art methods in this field often propose unifying the 2D video point tracking models, depth estimation, and camera pose estimation methods, *e.g.*, DELTA [18], TAPIP3D [19], and SpatialTracker [20]. We note that these models focus on motion estimation in videos rather than learning holistic prior for 3D shapes, and further discuss the 3D point mapping methods and their extension to dynamic scenes in the follwing subsection.

2.3 3D point mapping models

Since the seminal work in DUSt3R [2], which introduced the Siamese decoding architecture for the pair-wise point mapping, numerous contributions have been made to enhance the generalization performance to various 3D modeling scenarios, namely the application to dynamic scenes, and the extended temporal window size for processing multiple frames.

For example, the approaches for fine-tuning from synthetic data is found effective for enhancing the 3D reconstruction in dynamic scenes [3, 21], although they require an external prior models for motion estimation or metric depth estimation. Later, the extended task definitions have been introduced to directly learn priors for the temporal correspondence, or tracking between a fair of input frames [22, 23]. However, these methods are still constrained to the pair-wise architecture and often rely on heavy test-time optimization for processing multiple frames. Although the technique for expediting the optimization is a concurrent area of research [24], the time cost persists to be high.

In order to achieve the multi-frame modeling without relying on the optimization, new architectures have been proposed, such as the memory bank [4, 25], the multi-view cross attention [26], non-Siamese decoders [27], and employing DINO [28] features as additional inputs [29]. However, the task considered by these models cannot explicitly handle motion in video frames, which hinders generalization to dynamic videos. While our framework also considers an enhanced architecture with the modified attention, we additionally generalize the task for a joint point mapping and trajectory prediction, which has significant effect for understanding dynamic scenes.

It is also worth noting concurrent works such as GFlow [30], POMATO [31], St4RTrack [32], which have been devoted to tackle handling dynamic videos in point mapping architectures. For example, GFlow [30] employs the Gaussian Splatting [33] for better optimization, POMATO [31] proposes a new temporal prediction head for motion estimation, and St4RTrack [32] employs a pair-wise tracking task and architecture.

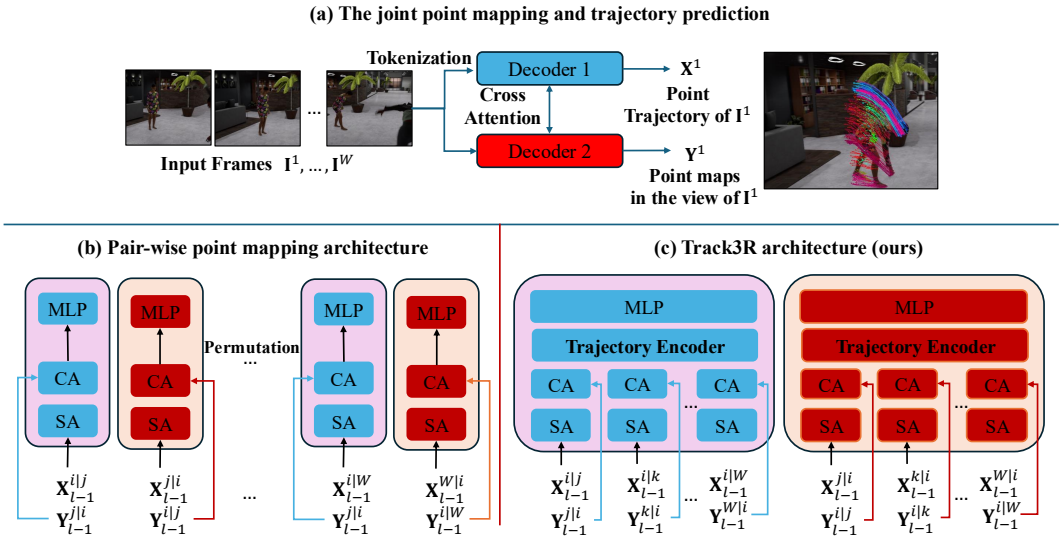

Figure 2: **Illustration of the prediction pipeline in Track3R.** The top figure (a) depeicts the overall pipeline, with the first decoder (Decoder 1) and the second decoder (Decoder 2), shared by both the point mapping frameworks. The bottom-left figure (b) illustrates the design of the decoder blocks in the pair-wise point mapping, using the self-attention (SA) and the cross-attention (CA) mechanisms. The bottom-right figure (c) illustrates our architecture for joint point mapping and trajectory, equipped with the proposed trajectory encoder.

3 Method

In this section, we provide the details of our architecture design for point mapping and trajectory prediction given a video sequence. We denote scalars using normal letters, and tensors using bold letters with a superscript denoting frame indices. For example, an input RGB video frame is $\mathbf{I}^i \in \mathbb{R}^{U\times V\times 3}$, where $U \times V$ is the resolution, and a frame tokenization is $\mathbf{F}^i \in \mathbb{R}^{N\times D}$, where $N = \frac{U}{P} \times \frac{V}{P}$ with the patch size P and the embedding dimension D. Tensors can be indexed, such as $\mathbf{F}^i(n) \in \mathbb{R}^D$, where $\mathbf{F}^i \equiv [\mathbf{F}^i(1), ..., \mathbf{F}^i(N)]$. Finally, to highlight that a feature or data for frame i is conditioned on the frame j, we use the superscript $i|j$, such as the point map $\mathbf{X}^{i|j} \in \mathbb{R}^{U\times V\times 3}$.

3.1 Pair-wise point mapping

Given a pair of frames $(\mathbf{I}^i, \mathbf{I}^j)$, the point mapping model aims to predict 2 different 3D points: the first decoder predicting $\mathbf{X}^{i|j}$ which represents the 3D coordinate of $\mathbf{I}^i$, and the second decoder predicting $\mathbf{Y}^{j|i}$ which represents the 3D coordinate of $\mathbf{I}^j$ in the camera view of $\mathbf{I}^i$. Specifically, the vanilla Siamese architecture [2] employs transformer blocks with cross-attention. To describe the operations within a block, we denote the tokens in the first decoder as $\mathbf{X}_l^{i|j}$, and the second decoder as $\mathbf{Y}_l^{i|j}$, where $l \in \{0, ..., L\}$ is the block index, with the initialization $\mathbf{X}_0^{i|j} := \mathbf{F}^i$ and $\mathbf{Y}_0^{j|i} := \mathbf{F}^j$.

In each transformer block (Figure 2b), the cross-attention $\texttt{CA}(\cdot;\cdot)$, placed next to the self-attention $\texttt{SA}(\cdot)$, conveys information between the two decoders, followed by the $\texttt{MLP}(\cdot)$ layer, producing the parallel outputs,

$$\tilde{\mathbf{X}}_l^{i|j} := \texttt{CA}\big(\texttt{SA}(\mathbf{X}_{l-1}^{i|j}); \mathbf{Y}_{l-1}^{j|i}\big) \quad \text{and} \quad \tilde{\mathbf{Y}}_l^{j|i} := \texttt{CA}\big(\texttt{SA}(\mathbf{Y}_{l-1}^{j|i}); \mathbf{X}_{l-1}^{i|j}\big) \tag{1}$$

$$\mathbf{X}_l^{i|j} := \texttt{MLP}\big(\tilde{\mathbf{X}}_l^{i|j}\big) \quad \text{and} \quad \mathbf{Y}_l^{j|i} := \texttt{MLP}\big(\tilde{\mathbf{Y}}_l^{j|i}\big), \tag{2}$$

assuming the skip-connections [34, 35] existing in the layers. To produce the final outputs, the DPT head layer [36] is employed, which takes these parallel block-wise tokens as the input,

$$\mathbf{X}^{i|j} := \texttt{Head}\big(\mathbf{X}_0^{i|j}; \mathbf{X}_1^{i|j}; ...; \mathbf{X}_L^{i|j}\big) \quad \text{and} \quad \mathbf{Y}^{j|i} := \texttt{Head}\big(\mathbf{Y}_0^{j|i}; \mathbf{Y}_1^{j|i}; ...; \mathbf{Y}_L^{j|i}\big). \tag{3}$$

Although we abuse the same notations SA, CA, MLP, and Head for the two decoders and for all block indices $l \in \{1, ..., L\}$, we note that their weight parameters are all different.

This pair-wise point mapping model is typically executed twice, for the permutation of the input frames, which enables downstream tasks, such as 2-view geometry, estimating camera poses, etc. When processing a larger number of frames, *e.g.*, $W > 2$, the inference is performed over $\texttt{Permutation}(W, 2)$ for all $i \in \{1, ..., W\}$ and $j \in \{1, ..., W\}$.

3.2 Generalized point trajectory prediction

As discussed in the previous section, the point mapping on W frames would essentially produce two sets of point map sequences per frame $\mathbf{X}^W \in \mathbb{R}^{(W-1)\times U\times V\times 3}$ and $\mathbf{Y}^W \in \mathbb{R}^{(W-1)\times U\times V\times 3}$. For example, assuming $W = 3$ input frames, $\{\mathbf{I}^1, \mathbf{I}^2, \mathbf{I}^3\}$, $\mathbf{I}^3$ would be associated with the set of point maps from the first and second decoders, respectively:

$$\mathbf{X}^3 := [\mathbf{X}^{3|1}, \mathbf{X}^{3|2}] \quad \text{and} \quad \mathbf{Y}^3 := [\mathbf{Y}^{1|3}, \mathbf{Y}^{2|3}]. \tag{4}$$

Given these sequences, one can quickly notice that the outputs from the first decoder, $\mathbf{X}^{3|1}$ and $\mathbf{X}^{3|2}$, represent the same semantics: the 3D coordinate of $\mathbf{I}^3$ in the camera view of $\mathbf{I}^3$. Our key idea is to resolve this duplication, by considering a modified definition, $\mathbf{X}^{3|1}$ and $\mathbf{X}^{3|2}$ representing the 3D coordinate of $\mathbf{I}^3$ in the frame index of $\mathbf{I}^1$ and $\mathbf{I}^2$, respectively, in the fixed camera view of $\mathbf{I}^3$. That is, we predict the 3D motion trajectory of the frame $\mathbf{I}^3$.

In order to learn this new task, we employ the confidence-aware regression loss from the original point mapping literature [2], yet provide the 3D trajectory annotations as the true targets for $\mathbf{X}$'s, which have been relatively sparse in datasets considered by prior art. Therefore, we further facilitate the learning by utilizing human motion dataset, which provide a denser motion trajectory and human body models [9], where the corresponding video frames can be readily rendered via a dedicated framework [10]. We note that the task definition of $\mathbf{Y}$'s remain unchanged, representing the 3D coordinate of a frame pixels mapped to the views of other frames.

3.3 Trajectory encoder

Despite the generalized task, the pair-wise architecture is still constrained to process the complex dynamics spanning $W > 2$. To tackle this problem, we describe our method to jointly process multiple frames (*i.e.*, $W > 2$). Specifically, we enable it with the trajectory encoder module. This module is based on the factorized temporal attention, which collects the tokens in the same spatial index over the frames to encode the inter-frame dynamics back to each token.

Let us consider the frame $\mathbf{I}^W$, paired with others $\{\mathbf{I}^1, ..., \mathbf{I}^{W-1}\}$ and their corresponding tokens within the intermediate cross-attention stage of the decoder blocks in Equation (1),

$$\tilde{\mathbf{X}}_l^{W|\{k<W\}} = \{\tilde{\mathbf{X}}_l^{W|1}, ..., \tilde{\mathbf{X}}_l^{W|W-1}\} \quad \text{and} \quad \tilde{\mathbf{Y}}_l^{W|\{k<W\}} = \{\tilde{\mathbf{Y}}_l^{W|1}, ..., \tilde{\mathbf{Y}}_l^{W|W-1}\}. \tag{5}$$

Intuitively, gathering from a same spatial index, *e.g.*, a stack of tokens $[\tilde{\mathbf{Y}}_l^{W|1}(n), ..., \tilde{\mathbf{Y}}_l^{W|W-1}(n)]$ by indexing each element in Equation (5), can represent the spatio-temporal dynamics of the patch region represented by $\mathbf{F}^W(n)$. Therefore, projecting this feature onto each token can encode the dynamics. Specifically, we apply a factorized attention[2] with causal masks to implement the function, coined trajectory attention $\texttt{TA}(\cdot;\cdot)$,

$$\bar{\mathbf{X}}_l^{W|j} := \texttt{TA}(\tilde{\mathbf{X}}_l^{W|j}; \tilde{\mathbf{X}}_l^{W|\{k<W\}}) \quad \text{and} \quad \bar{\mathbf{Y}}_l^{W|j} := \texttt{TA}(\tilde{\mathbf{Y}}_l^{W|j}; \tilde{\mathbf{Y}}_l^{W|\{k<W\}}), \tag{6}$$

where

$$\bar{\mathbf{X}}_l^{W|j}(n) = \texttt{CA}\big(\tilde{\mathbf{X}}_l^{W|j}(n); [\tilde{\mathbf{X}}_l^{W|1}(n), ..., \tilde{\mathbf{X}}_l^{W|j}(n)]\big), \tag{7}$$

$$\bar{\mathbf{Y}}_l^{W|j}(n) = \texttt{CA}\big(\tilde{\mathbf{Y}}_l^{W|j}(n); [\tilde{\mathbf{Y}}_l^{W|1}(n), ..., \tilde{\mathbf{Y}}_l^{W|j}(n)]\big). \tag{8}$$

[2]We adjust the relative position embedding to encode a spatial index with the size $D/2$, and a time index with the size $D/2$.

However, naively inserting this layer to each decoder block of a pre-trained Siamese model results in sub-optimal performance after training on dynamic scenes. In fact, prior art finds that retaining strong 3D prior learned from static datasets is crucial for dynamic scenes [3]. Since the trajectory attention deviates the computation graph of a pair-wise model, the model can lose the pre-trained 3D prior. We note that it is also non-trivial to pre-train a multi-frame model from scratch, since the training data for 3D geometry is often a pair of images [2], rather than a video stream data.

To address the problem, we aim to minimize the effect of modification in the initial state of the model. Specifically, inspired by model inflation techniques in video transformers [8, 37], which maintain image prior by attenuating the activation of the temporal attentions, we introduce the layerscale $\texttt{LS}(\cdot)$ initialized to a very small scalar [38] to the module, referring to the whole layer as the trajectory encoder $\texttt{TE}(\cdot;\cdot)$,

$$\begin{aligned}\bar{\mathbf{X}}_l^{W|j} &:= \texttt{TE}\big(\tilde{\mathbf{X}}_l^{W|j}; \tilde{\mathbf{X}}_l^{W|\{k<W\}}\big) \\ &:= \tilde{\mathbf{X}}_l^{W|j} + \texttt{LS}\big(\texttt{TA}(\tilde{\mathbf{X}}_l^{W|j}; \tilde{\mathbf{X}}_l^{W|\{k<W\}})\big), \end{aligned} \tag{9}$$

$$\begin{aligned}\bar{\mathbf{Y}}_l^{W|j} &:= \texttt{TE}\big(\tilde{\mathbf{Y}}_l^{W|j}; \tilde{\mathbf{Y}}_l^{W|\{k<W\}}\big) \\ &:= \tilde{\mathbf{Y}}_l^{W|j} + \texttt{LS}\big(\texttt{TA}(\tilde{\mathbf{Y}}_l^{W|j}; \tilde{\mathbf{Y}}_l^{W|\{k<W\}})\big). \end{aligned} \tag{10}$$

This design ensures that the model is equivalent to the pair-wise architecture in the initial state, retaining the pre-trained 3D prior.

To summarize, we enhance the model architecture for multi-frame, and generalize the task definition (in Section 3.2) of the corresponding outputs. Throughout the training on dynamic scenes, our model initialize from a strong 3D shape prior, and gradually learn to model complex multi-frame dynamics and predict 3D motion estimation, achieving the joint point mapping and trajectory prior.

4 Experiment

In this section, we present the experimental details and compare Track3R to state-of-the-art baselines. In Section 4.1, we provide the training details, such as the training dataset, schedules, and hyperparameters. In Section 4.2, we discuss the experimental details, such as the baselines, checkpoints, and the inference configurations. Next, we provide the results on the joint point mapping and trajectory prediction in Section 4.3, which aims to evaluate the quality of the motion and shape prior learned by Track3R in dynamic scenes. Then, we further study the downstream task, *e.g.*, the feed-forward camera pose estimation in Section 4.4, which aims to compare the ability to disentangle the camera motion in dynamic scenes. Finally, we present the ablation study of the proposed method in Section 4.5.

4.1 Training details

We initialize the Track3R with the pre-trained weight published by MonST3R [3], a pair-wise point mapping architecture trained on dynamic scenes covered by synthetic datasets [12, 39, 40], starting from the pre-trained DUSt3R [2], which is trained on 8M images that capture the real-world static scenes at scale, such as ScanNet [41] and StaticThings3D [42].

For the fine-tuning, we employ the 3D trajectories and video frames, the combination of SMPL-X human motion trajectory [9] and the associated video frames rendered with BEDLAM [10], along with the publicly available Waymo [43] dataset. Specifically, we supervise the output X from the first decoder with the 3D trajectory annotation, and supervise the output Y from the second decoder with the ground truth depth maps, registered to the world coordinate system based on the annotated camera poses. We employ the confidence-aware and scale-invariant regression loss, following the baselines [2, 3].

We train Track3R for 25 epochs using the AdamW optimizer [44] with 25k clips of length $W = 6$ per epoch, the mini-batch size 16, and the learning rate 1×10^{-4}. The training takes approximately 36 hours on the system equipped with 8 nVidia A100 GPUs.

Table 1: **Joint 3D motion and shape performance.** The quality of 3D motion estimation (EPE_{3D}) and the point mapping (Abs-rel) are compared among Track3R (ours) and the baselines. The baselines that perform additional test-time optimizations are grouped in the top block, and the baselines for feed-forward prediction are grouped in the bottom block. The best scores are highlighted with bold numbers, and the best scores within a model category are highlighted with underlines.

		iPhone		Sintel		Point Odyssey	
Method	FPS	EPE_{3D}	Abs-rel	EPE_{3D}	Abs-rel	EPE_{3D}	Abs-rel
DUSt3R (2024)	0.62	0.973	1.212	0.565	0.422	0.933	0.184
MonST3R (2025)	0.41	0.619	0.310	0.401	0.335	0.481	0.090
MegaSaM (2025)	0.97	0.598	**0.211**	0.372	**0.231**	0.455	0.091
Align3R (2025)	0.17	0.581	0.290	0.487	0.263	0.670	**0.075**
Fast3R(2025)	91.81	0.692	0.419	0.667	0.517	0.771	0.214
Spann3R(2025)	20.77	0.658	0.431	0.523	0.622	0.591	0.231
CUT3R(2025)	27.43	0.701	0.408	0.681	0.428	0.609	0.177
Track3R (ours)	23.81	**0.431**	0.344	**0.312**	0.374	**0.399**	0.165

Table 2: **Feed-forward camera pose estimation.** The quality of estimating camera translation (ATE and RPE-t) and rotation (RPE-r) are comapred among the feed-forward prediction baselines. The best scores are highlighted with bold numbers, and the second-best scores are highlighted with underlines.

	iPhone			Sintel			TUM-dynamic		
Method	ATE	RPE-t	RPE-r	ATE	RPE-t	RPE-r	ATE	RPE-t	RPE-r
Fast3R (2025)	0.413	0.294	1.561	0.377	0.150	3.233	0.129	0.111	2.794
Spann3R (2025)	0.552	0.310	2.301	0.329	0.110	4.471	0.056	0.021	0.591
CUT3R (2025)	0.291	0.184	0.848	0.213	**0.064**	**0.596**	**0.046**	**0.015**	**0.473**
Track3R (ours)	**0.233**	**0.150**	**0.694**	**0.201**	0.066	0.621	0.051	0.020	0.517

4.2 Experimental details

We consider 7 different baseline point mapping models, DUSt3R [2], MonST3R [3], MegaSaM [24], Align3R [21], Fast3R [27], Spann3R [25], and CUT3R [4]. We experiment with the checkpoint provided in the official open-source repository hosted by their authors, following the default image processing in each model, *e.g.*, , the input dimensions are, the longer side length of 512 in DUSt3R [2], MonST3R [3], Align3R [21], and Fast3R [27], the longer side length of 672, in MegaSaM [24], and, the square 256×256 in Spann3R [25].

Unless otherwise specified, we always choose the temporal window size of the inference $W = 6$ for evaluating our method and the baselines. We note that the pair-wise processing baselines are iteratively executed to match the required window size.

To evaluate the feed-forward camera pose estimation in Section 4.4, we employ the weighted Procrustes solver to derive the relative rotation and translation between the frames, and the weighted least squares solver to estimate the camera intrinsic parameters, similar to the experimental configuration in CUT3R [4].

4.3 Joint point mapping and 3D motion estimation.

In this section, we evaluate the joint point mapping and 3D motion estimation quality. We employ 3 different test datasets covering dynamic scenes: iPhone dataset [45], Point Odyssey [12], and Sintel [46]. iPhone dataset covers the real-world scene, which originally provides monocular video frames, depth, and camera poses collected with the Lidar and IMU sensors [45]. To obtain the 3D motion trajectory annotation, we utilize the track optimization part of the sophisticated stereo video optimization framework [22]. Point Odyssey and Sintel are synthetic datasets rendered with the 3D engine, which provides the ground truth depth and 3D motion trajectories. We note that the validation sets are utilized for these datasets.

Table 3: **Ablation study.** The effect of trajectory encoder, the joint training objective, and the training with human motion dataset are studied in terms of the quality of 3D motion estimation (EPE_{3D}) and the point mapping (Abs-rel).

	iPhone		Sintel		Point Odyssey	
Modules	EPE_{3D}	Abs-rel	EPE_{3D}	Abs-rel	EPE_{3D}	Abs-rel
Base Model	0.803	0.879	0.695	0.737	0.980	0.281
+ Trajectory Encoder	0.682	0.593	0.576	0.531	0.704	0.239
+ Joint Objective	0.445	0.357	0.429	0.396	0.463	0.181
+ Human Motion	**0.431**	**0.344**	**0.312**	**0.374**	**0.399**	**0.165**

To provide a holistic view over both 3D motion and shape qualities, we borrow metrics from literature for each task: the 3D end-point-error (EPE_{3D}; a regression quality of 3D motion trajectory) [47] and the absolute relative error (Abs-rel; the accuracy of point mapping to be within a 1.25-factor of the ground truth) [4]. We compare Track3R and the baselines: DUSt3R [2], MonST3R [3], MegaSaM [24], Align3R [21], Spann3R [25], Fast3R [27], CUT3R [4] in Table 1.

To begin with, Track3R demonstrates the strongest results in terms of the quality of 3D motion estimation (ECE_{3D}), *e.g.*, relative 25.8% improvement compared to Align3R [21] in iPhone dataset ($0.581 \rightarrow 0.431$). We observe the baselines employing test-time optimizations (upper block in Table 1) can demonstrate overall improved motion estimation performance than the feed-forward prediction methods (lower block in Table 1), except Track3R (ours).

Although the test-time optimization can reinforce overall consistency in point mapping along the frames, and inject external motion prior (*e.g.*, MonST3R [3]), it significantly deteriorates inference speed (FPS). Since our Track3R can directly learn the 3D motion prior during training, instead of the test-time, it can achieve a better motion estimation performance and provides a reasonable inference speed as well, *e.g.*, 23.81 (FPS).

In terms of the point mapping accuracy (Abs-rel), Track3R consistently demonstrates the strongest results among the feed-forward prediction methods, *e.g.*, relative 15.7% improvement compared to CUT3R [4] in iPhone dataset ($0.408 \rightarrow 0.344$). Although there is a gap between the test-time optimizations and ours, the significant improvement compared to the strongest feed-forward baseline (CUT3R [4]) supports the effect of the joint prior learning by our method.

4.4 Downstream tasks

In this section, as the downstream application, we evaluate the feed-forward camera pose estimation in dynamic scenes, and provide visualization of the point maps in the world coordinate system.

First, in the feed-forward camera pose estimation, we consider the benchmark with respect to Sintel [46] and TUM-dynamics [48], following the configuration in CUT3R [4] for dynamic scenes, and also the customized iPhone dataset [45] evaluated with the sensory camera poses. In Table 2, we compare Track3R (ours) and the feed-forward prediction baselines: Fast3R [27], Spann3R [25], and CUT3R [4]. Our method achieves the strongest results in iPhone dataset, and a performance comparable to CUT3R [4] on in Sintel and TUM-dynamics datasets. For example, Track3R can demonstrate significant improvements in iPhone dataset, which supports that the 3D prior learn by our method can provide a robust prior that better generalize to the real-world dynamic scenes. We observe that CUT3R [4] particularly performs well on synthetic datasets, which is possible by the pose prediction head trained with the ground truth poses in synthetic datasets. In the extension of our goal to learn joint 3D priors, employing such an idea into Track3R can be interesting future work.

Next, for a qualitative demonstration, we provide the visualization of the 3D points in Figure 3, executed on DAVIS video frames [5]. We find our method tends to demonstrate more consistent point maps over the frames, *e.g.*, the background objects and the scene are consistently depicted, comparing the pair-wise point mapping baseline [3] (with red boxes) and our method performing joint point mapping and trajectory prediction (with blue boxes). This supports the significance of our method, which facilitates learning useful representation for predicting accurate point maps as well, even if affected by complex dynamic scenes.

Figure 3: **Qualitative comparison of 3D points in dynamic scenes.** The 3D points predicted the pair-wise point mapping baseline [3] and Track3R (ours) are compared, using DAVIS video samples [5]. Dynamic regions are highlited with the red and blue boxes.

4.5 Ablation study

In this section, we conduct ablation study of the proposed techniques, namely the trajectory encoder, the joint point mapping and trajectory prediction, and the training with human motion data, comparing EPE_{3D} (3D motion estimation) and Abs-rel (point mapping) in Table 3. Overall, we find employing each module is significant to the performance of Track3R, where the flexibility to process multi-frame dynamics is enabled by the trajectory encoder, a robust prior learning is achieved via the joint point mapping and motion estimation objective, and additional performance boosts are observed when trained on human motion.

Although the pair-wise architecture [2, 3] can produce pointmaps for more than 2 frames by executing multiple pair-wise inferences, its design inevitably enforces the assumption that the distributions of consecutive pointmaps are independent. For example, given $\{\mathbf{I}^i, \mathbf{I}^j, \mathbf{I}^k\}$, a pair-wise model assumes that a joint density $\texttt{Pr}(\mathbf{Y}^{i|j}, \mathbf{Y}^{i|k}, \mathbf{Y}^{j|k})$ is proportional to $\texttt{Pr}(\mathbf{Y}^{i|j}) \cdot \texttt{Pr}(\mathbf{Y}^{i|k}) \cdot \texttt{Pr}(\mathbf{Y}^{j|k})$. However, in practice, including the scenarios represented by our evaluation, there exists an extreme case where $\mathbf{I}^i$ and $\mathbf{I}^k$ are completely non-overlapping, so that the pair-wise model assigns an erroneous estimate of $\texttt{Pr}(\mathbf{Y}^{i|k})$, which can induce significant failure modes of estimating the joint density. Since Track3R can relax this constraint for multiple frames, it can learn the joint point mapping and trajectory prior that is more close to the true nature of the dynamic scenes.

5 Discussion

In this section, we discuss the limitation of Track3R and the future research directions in Section 5.1. We also discuss the potential negative societal impact in Section 5.2.

5.1 Limitation

Despite the promising results demonstrated by Track3R, the scarcity of dynamic scenes for training can hinder the generalization performance. To mitigate the problem, we employ the human motion dataset for training. However, the synthesized video inputs can make a distribution shift in the visual texture learned in a pre-trained model. Therefore, designing new training datasets, self-supervised learning with unlabeled data, or an objective functions robust to the distribution shift can be interesting future directions. It is also worth noting that we focus on the realistic scenarios where the observation is captured by a monocular video camera, rather than multiple synchronized cameras capturing one scene. Although it would be straightforward to apply Track3R for the synchronized cameras, we believe that there is a room to exploit useful properties, such as epipolar geometry [49] of the synchronized cameras, which is another interesting future direction.

5.2 Potential negative societal impact

While the joint point mapping and motion estimation by Track3R can be beneficial for various video understanding applications, such as novel-view synthesis, depth estimation, and action recognition, the emergence of unexpected behavior within Track3R can lead to misrepresentations of the real video data. For those applications that require extremely accurate models for safety-related judgments, such as depth estimation for autonomous driving, the unexpected behaviors must be carefully managed. To ensure the reliability of systems using point tracking predictions, we recommend to conduct thorough investigations and implement robust mitigation strategies to minimize potential risks, thereby increasing the overall safety and effectiveness of these applications.

6 Conclusion

In this paper, we propose Track3R, a joint point mapping and motion estimation framework for learning holistic 3D priors dynamic scenes. We tackle the limitations in existing point mapping baselines, sub-optimal under complex dynamic scenes. For example, we propose to encode the dynamics of the 3D points over multiple frames beyond the pairs, and generalize the task definition to predict the 3D motion trajectories, as well as the static point maps. Our method significantly improves the expressiveness of the model architecture for dynamic scenes, and enables predicting the disentangled representation of the 3D motion and shapes. In the experiments, we find our method can outperform the baselines, for both the 3D motion estimation task and the point mapping task. Overall, our work highlights the effectiveness of jointly solving 3D geometry and motion tasks, and we believe our work could inspire researchers to further leverage it in the future.

Acknowledgements. This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.RS-2019-II190075 Artificial Intelligence Graduate School Program(KAIST); No. RS2024-00509279, Global AI Frontier Lab), the National Supercomputing Center with supercomputing resources including technical support KSC-2025-CRE-0435, and RLWRLD, Inc.

References

[1] Johannes Kopf, Xuejian Rong, and Jia-Bin Huang. Robust consistent video depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1611–1621, 2021. 1

[2] 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. 1, 3, 4, 5, 6, 7, 8, 10

[3] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. In *International Conference on Learning Representations*, 2025. 1, 2, 3, 6, 7, 8, 9, 10

[4] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *CVPR*, 2025. 1, 3, 7, 8

[5] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Computer Vision and Pattern Recognition*, 2016. 2, 8, 9

[6] Philippe Weinzaepfel, Vincent Leroy, Thomas Lucas, Romain Brégier, Yohann Cabon, Vaibhav Arora, Leonid Antsfeld, Boris Chidlovskii, Gabriela Csurka, and Jérôme Revaud. Croco: Self-supervised pre-training for 3d vision tasks by cross-view completion. *Advances in Neural Information Processing Systems*, 35:3502–3516, 2022. 2

[7] Philippe Weinzaepfel, Thomas Lucas, Vincent Leroy, Yohann Cabon, Vaibhav Arora, Romain Brégier, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jérôme Revaud. CroCo v2: Improved Cross-view Completion Pre-training for Stereo Matching and Optical Flow. In *ICCV*, 2023. 2

[8] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, page 4, 2021. 2, 6

[9] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 10975–10985, 2019. 2, 5, 6

[10] Michael J. Black, Priyanka Patel, Joachim Tesch, and Jinlong Yang. BEDLAM: A synthetic dataset of bodies exhibiting detailed lifelike animated motion. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 8726–8737, June 2023. 2, 5, 6

[11] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3749–3761, 2022. 2

[12] Yang Zheng, Adam W. Harley, Bokui Shen, Gordon Wetzstein, and Leonidas J. Guibas. Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In *ICCV*, 2023. 2, 6, 7

[13] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2021. 3

[14] David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19457–19467, 2024. 3

[15] Qianqian Wang, Vickie Ye, Hang Gao, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4d reconstruction from a single video. *arXiv preprint arXiv:2407.13764*, 2024. 3

[16] Carl Doersch, Ankush Gupta, Larisa Markeeva, Adria Recasens, Lucas Smaira, Yusuf Aytar, Joao Carreira, Andrew Zisserman, and Yi Yang. Tap-vid: A benchmark for tracking any point in a video. *Advances in Neural Information Processing Systems*, 35:13610–13626, 2022. 3

[17] Skanda Koppula, Ignacio Rocco, Yi Yang, Joe Heyward, Joao Carreira, Andrew Zisserman, Gabriel Brostow, and Carl Doersch. Tapvid-3d: A benchmark for tracking any point in 3d, 2024. *URL https://arxiv.org/abs/2407.05921*, 2(8):17. 3

[18] Tuan Duc Ngo, Peiye Zhuang, Evangelos Kalogerakis, Chuang Gan, Sergey Tulyakov, Hsin-Ying Lee, and Chaoyang Wang. Delta: Dense efficient long-range 3d tracking for any video. In *The Thirteenth International Conference on Learning Representations*. 3

[19] Bowei Zhang, Lei Ke, Adam W Harley, and Katerina Fragkiadaki. Tapip3d: Tracking any point in persistent 3d geometry. *arXiv preprint arXiv:2504.14717*, 2025. 3

[20] Yuxi Xiao, Qianqian Wang, Shangzhan Zhang, Nan Xue, Sida Peng, Yujun Shen, and Xiaowei Zhou. Spatialtracker: Tracking any 2d pixels in 3d space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20406–20417, 2024. 3

[21] Jiahao Lu, Tianyu Huang, Peng Li, Zhiyang Dou, Cheng Lin, Zhiming Cui, Zhen Dong, Sai-Kit Yeung, Wenping Wang, and Yuan Liu. Align3r: Aligned monocular depth estimation for dynamic videos. *arXiv preprint arXiv:2412.03079*, 2025. 3, 7, 8

[22] Linyi Jin, Richard Tucker, Zhengqi Li, David Fouhey, Noah Snavely, and Aleksander Holynski. Stereo4d: Learning how things move in 3d from internet stereo videos. *arXiv preprint*, 2024. 3, 7

[23] Edgar Sucar, Zihang Lai, Eldar Insafutdinov, and Andrea Vedaldi. Dynamic point maps: A versatile representation for dynamic 3d reconstruction. *arXiv preprint arXiv:2503.16318*, 2025. 3

[24] Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast, and robust structure and motion from casual dynamic videos. *arXiv preprint arXiv:2412.04463*, 2025. 3, 7, 8

[25] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. *arXiv preprint arXiv:2408.16061*, 2025. 3, 7, 8

[26] Yuzheng Liu, Siyan Dong, Shuzhe Wang, Yingda Yin, Yanchao Yang, Qingnan Fan, and Baoquan Chen. Slam3r: Real-time dense scene reconstruction from monocular rgb videos. *arXiv preprint arXiv:2412.09401*, 2024. 3

[27] 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)*, June 2025. 3, 7, 8

[28] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024, 2024. URL https://openreview.net/forum?id=a68SUt6zFt. 3

[29] 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. 3

[30] Shizun Wang, Xingyi Yang, Qiuhong Shen, Zhenxiang Jiang, and Xinchao Wang. Gflow: Recovering 4d world from monocular video. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 7862–7870, 2025. 3

[31] Songyan Zhang, Yongtao Ge, Jinyuan Tian, Guangkai Xu, Hao Chen, Chen Lv, and Chunhua Shen. Pomato: Marrying pointmap matching with temporal motion for dynamic 3d reconstruction. *arXiv preprint arXiv:2504.05692*, 2025. 3

[32] Haiwen Feng, Junyi Zhang, Qianqian Wang, Yufei Ye, Pengcheng Yu, Michael J Black, Trevor Darrell, and Angjoo Kanazawa. St4rtrack: Simultaneous 4d reconstruction and tracking in the world. 2025. 3

[33] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 3

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[35] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[36] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 4

[37] Mandela Patrick, Dylan Campbell, Yuki M. Asano, Ishan Misra Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and Joao F. Henriques. Keeping your eye on the ball: Trajectory attention in video transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 6

[38] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 32–42, 2021. 6

[39] Lukas Mehl, Jenny Schmalfuss, Azin Jahedi, Yaroslava Nalivayko, and Andrés Bruhn. Spring: A high-resolution high-detail dataset and benchmark for scene flow, optical flow and stereo. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 6

[40] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020. 6

[41] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12–22, 2023. 6

[42] Philipp Schröppel, Jan Bechtold, Artemij Amiranashvili, and Thomas Brox. A benchmark and a baseline for robust multi-view depth estimation. In *2022 International Conference on 3D Vision (3DV)*, pages 637–645. IEEE, 2022. 6

[43] 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. 6

[44] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7. 6

[45] Hang Gao, Ruilong Li, Shubham Tulsiani, Bryan Russell, and Angjoo Kanazawa. Monocular dynamic view synthesis: A reality check. *Advances in Neural Information Processing Systems*, 35:33768–33780, 2022. 7, 8

[46] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, October 2012. 7, 8

[47] Zachary Teed and Jia Deng. Raft-3d: Scene flow using rigid-motion embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8375–8384, 2021. 8

[48] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 573–580. IEEE, 2012. 8

[49] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 10

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 [NA] .
- [NA] 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 NA).

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 be asked to also 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 "[NA] " 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: All claims in the introduction and abstract accurately reflect the contribution and scope, which are then verified in Section 4.

   Guidelines:

   - The answer NA 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 NA 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: Section 5 discusses it.

Guidelines:

- The answer NA 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: [NA]

   Justification: We do not have a theory in this paper.

   Guidelines:

   - The answer NA 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: We have included the implementation details in Section 4.1 and Section 4.2.

   Guidelines:

   - The answer NA 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: We plan to release workspaces related to data generation and learning to the public. We also utilize open source models for training.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.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://nips.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, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide the detail of the training/evaluation setup, dataset, and hyperparameters in Section 4.1 and Section 4.2.

   Guidelines:

   - The answer NA 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: [No]

   Justification: All experiments are conducted with the same and commonly used random seed.

   Guidelines:

   - The answer NA 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: We provide the computer resources for training Section 4.2.

   Guidelines:

   - The answer NA 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: We do not have any ethical concerns regarding the paper.

   Guidelines:

   - The answer NA 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: [Yes]

    Justification: We discuss the societal impact in Section 5

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA 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., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our method does not introduce risks for misuse.

Guidelines:

- The answer NA 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: [Yes]

    Justification: We have cited all papers and datasets used in the paper.

    Guidelines:

    - The answer NA 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: [No]

    Justification: We will release the Pytorch implementation after the acceptance.

    Guidelines:

    - The answer NA 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: [NA]

Justification: We use existing benchmark datasets and do not have any crowdsourcing datasets or experiments in the paper.

Guidelines:

- The answer NA 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: [NA]

Justification: We do not have human subject in the research.

Guidelines:

- The answer NA 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 rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development does not involve LLMs.

Guidelines:

- The answer NA 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 (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.