Dynamic Camera Poses and Where to Find Them

Chris Rockwell[1,2] Joseph Tung[3] Tsung-Yi Lin[1]
Ming-Yu Liu[1] David F. Fouhey[3] Chen-Hsuan Lin[1]
NVIDIA[1] University of Michigan[2] New York University[3]
https://research.nvidia.com/labs/dir/dynpose-100k

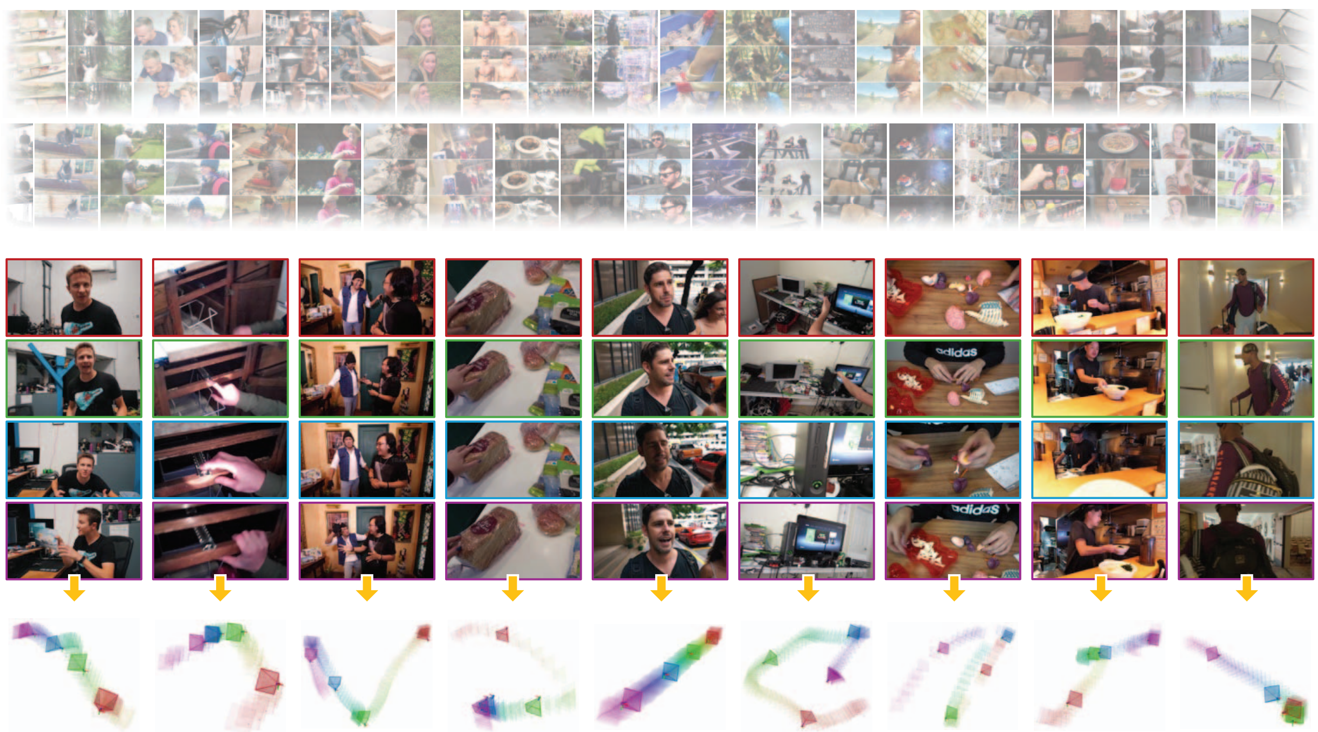

Figure 1. We introduce **DynPose-100K**, a large-scale video dataset of dynamic content with camera annotations. DynPose-100K consists of 100,131 Internet videos that span diverse settings. We curate DynPose-100K such that videos contain dynamic content while ensuring the cameras are able to be estimated (including intrinsics and poses). Towards this end, we address two challenging problems: (a) identifying the videos suitable for camera estimation, and (b) improving the camera estimation algorithm for dynamic videos.

Abstract

Annotating camera poses on dynamic Internet videos at scale is critical for advancing fields like realistic video generation and simulation. However, collecting such a dataset is difficult, as most Internet videos are unsuitable for pose estimation. Furthermore, annotating dynamic Internet videos present significant challenges even for state-of-the-art methods. In this paper, we introduce DynPose-100K, a large-scale dataset of dynamic Internet videos annotated with camera poses. Our collection pipeline addresses filtering using a carefully combined set of task-specific and generalist models. For pose estimation, we combine the latest techniques of point tracking, dynamic masking, and structure-from-motion to achieve improvements over the state-of-the-art approaches. Our analysis and experiments demonstrate that DynPose-100K is both large-scale and diverse across several key attributes, opening up avenues for advancements in various downstream applications.

1. Introduction

Annotating large-scale dynamic Internet video with camera information has the potential to advance many problems in computer vision and robotics. Such a dataset could power generative models [5, 7, 23, 31, 51, 51, 65, 90] to create camera-controlled dynamic videos [13, 29, 39, 52, 60, 85, 88, 97] and 4D scenes [2, 66, 99]. It could also enable large-scale training of view synthesis models [9, 16, 22, 46, 78, 86, 94] for extended reality applications. Additionally, this data could be transformational in robotics for tasks like imitation learning [1, 21, 54] or training in realistic simulation environments [6, 32, 84].

However, constructing a dataset of dynamic videos with camera annotations is challenging. First, most Internet videos are not suitable for camera pose estimation. They may be cartoons, contain substantial post-processing, or lack a clear frame of reference. Next, estimable videos are often low quality or do not contain scene dynamics. Furthermore, even if suitable target videos can be identified, the process of estimating dynamic camera poses remains challenging. Techniques that work well in static settings, such as structure-from-motion, struggle with moving objects, varying appearances, and other complex dynamics common in Internet videos.

Given the challenges, most existing datasets take a different approach. One line of work uses synthetic data [8, 26, 48, 49, 57, 100] where ground-truth camera data can be obtained. However, this often results in smaller-scale datasets (typically $<$500 videos) due to the high cost of 3D assets, and methods face a sim2real gap. Recent work finds camera data from real videos in restricted domains, such as egocentric videos [76], self-driving videos [70], hand-collected videos of pets [67] or specific actions [25]. These setups make SfM easier via dense viewpoints [76], LiDAR sensory data [70], multiple cameras [25, 34] and turntable-style videos [67]. However, such setups also impose significant constraints on the collected data.

We introduce DynPose-100K, a large collection of dynamic Internet videos with camera annotations. Sample videos and annotations appear in Figure 1. The dataset contains diverse content, a variety of dynamic object apparent sizes and targeted video length; and is analyzed in § 4. We address the central challenges of filtering for suitable dynamic video and annotating the corresponding camera poses with a carefully designed pipeline (§ 3). Our filtering pipeline combines a mix of specialized experts that handle common reasons videos are unsuitable, as well as a generalist VLM that can detect and remove a variety of issues. We produce accurate poses using a new method that merges the state-of-the-art in motion masking, correspondence tracking, and SfM. We apply this approach to Panda-70M [11, 89], filtering 3.2M videos to produce 100K with high-quality camera information.

Our contributions are summarized as:

1. We introduce DynPose-100K, the largest public dynamic Internet video dataset annotated with camera information. Analysis shows the dataset is diverse in content and dynamics, and has targeted video length.
2. We propose a filtering pipeline using specialist models and a VLM, motivated by Internet video analysis.
3. We propose a dynamic pose estimation pipeline integrating state-of-the-art components in tracking and masking.
4. Experiments show filtering selects videos with far higher precision than alternatives and pose estimation reduces error across metrics and settings by as much as 90%.

2. Related Work

This work aims to collect and annotate dynamic Internet video with camera pose. To do so, we face challenges in data curation and pose estimation.

Dynamic camera pose datasets. Due to challenges in predicting dynamic camera pose, datasets typically adopt strategies to replace or aid standard SfM. Synthetic datasets are appealing as ground truth pose can be obtained [8, 26, 48, 49, 57, 100, 100]. However, the requirement for engineered dynamic assets means the datasets are of small size and often contain limited dynamics *e.g.* objects flying [48] or falling [26]. In real-world video, SfM is made easier with *e.g.* turntable-style captures [67], dense viewpoints [76], multiple cameras [25], fisheye captures [34], and LiDAR [70]. We leverage the diverse Internet video dataset from Panda-70M [11] and show our method can annotate high quality poses without relying on the constraints of prior work. Concurrent works CamCo [88] and B-Timer [42] collect 12K and 40K diverse videos [3], respectively; we collect 100K and publicly release data.

Camera pose estimation. Given many overlapping views of a static scene [4, 10, 37, 45, 62, 101], classical SfM [61, 69, 71] and SLAM [50] are gold standards for precision, while learned methods recently are becoming competitive [20, 38, 68, 72, 75, 77, 80, 83, 93, 96] given large datasets [59, 82, 92, 101]. Dynamic Internet video is more challenging as dynamic correspondences cannot be used for bundle adjustment and also limit static correspondence. Recent works [24, 40, 47, 98] mask dynamics using learned prediction [12, 64, 98] or motion and semantic cues [28, 30, 73]. We upgrade masking using state-of-the-art methods in semantics, interaction, motion and tracking [14, 28, 33, 58, 73].

Correspondence estimation in Internet video is also challenging due to varying lighting and appearance [40]. As a result, we show masking+classical SfM [61] fails on diverse Internet video. ParticleSfM [98] improves by inducing dense correspondences via propagated optical flow [73]

Figure 2. **Panda-Test dataset statistics**. Statistics reflect human labels on held-out 1K video Panda-Test set, detailed in § 4.1. Only 9% are target dynamic camera pose estimation videos due to various issues, *e.g.* static scene, low-quality or non-real content, and ambiguous or blurry frame of reference. We focus on moving cameras to facilitate downstream tasks *e.g.* camera-controlled video generation and learned pose estimation. We remove unsuitable videos using a combination of specialist models and a generalist VLM.

and uses more robust Global Bundle Adjustment, Theia-SfM [71]. We adopt Theia-SfM and upgrade correspondence estimation by utilizing advances in long-term tracking [18, 27, 63, 79] via BootsTAP [19]. Concurrent work TracksTo4D [36] predicts pose and dynamic masks given 2D tracks [35] but focuses on small dynamics and circular camera motion [67]; we build off stronger 2D tracking BootsTAP and tackle diverse Internet video. Concurrent work DATAP-SfM [91] trains joint tracking and motion prediction on synthetic FlyingThings3D [48], we instead leverage tracking and masking methods trained on large amounts of real data [15, 19, 43, 58] and collect a large-scale dataset. Concurrent works MonST3R [95] and MegaSaM [41] generalize learned DUSt3R [81] and DROID-SLAM [74], respectively, to dynamic scenes; we build around classical SfM for pose annotation given its precision in longer video.

3. DynPose-100K: Dataset Curation

DynPose-100K is a large-scale video dataset comprising diverse dynamic Internet content with camera annotations. Assembling such a dataset is challenging due to two primary issues related to camera pose estimation: (a) the vast majority of Internet videos are unsuitable for successful camera pose estimation, and (b) camera pose estimation on dynamic videos is inherently a challenging problem. We address these challenges in this section.

3.1. Candidate Selection Criteria

Most videos in representative Internet video datasets such as Panda-70M [11] are unsuitable for estimating camera pose: Figure 2 shows only 9% of videos from a randomly selected test set of 1,000 videos from Panda-70M dataset satisfy the criteria for camera pose estimation in dynamic settings. We start by defining specific criteria to identify videos suitable for camera pose estimation, which we detect via subsequently defined data filtering steps.

We argue that videos should satisfy three criteria:

C1. **Real-world and quality video.** Videos must be from the real world *i.e.* not cartoons, computer screen recordings, side-by-side/composited videos, and heavily edited processed content. The videos should also be of sufficient quality, resolution, frame rate and should not have non-perspective distortion.

C2. **Feasibility for pose prediction.** Videos should not contain severe zoom-in or zoom-out effects, abrupt shot changes, or ambiguous reference frames (*e.g.* scenes filmed inside a moving car). Videos without static correspondences, *e.g.*, when the background is fully blurred or occluded should be excluded.

C3. **Dynamic camera and scene.** Non-static camera allows non-trivial pose for *e.g.* training pose estimation or camera-controlled video generation methods. Videos with moving cameras and dynamic scenes typically also provide richer data, *e.g.*, interactions involving people, supporting a wider range of downstream applications.

3.2. Candidate Video Selection

The filtering process aims to automatically select videos matching these three criteria. Our filtering pipeline identifies common reasons for unsuitability using specialized models (*e.g.* trained to detect distortion or estimate the camera focal length). On the other hand, a variety of other problems, *e.g.*, post-edited text are not effectively handled by these specialized models. To this end, we use the recent vision-language models (VLM) [56] and prompt them to detect a variety of issues. As will be shown in § 4.1, while individual filters (including the VLM) are insufficient on their own, their combined use is effective.

Filtering using task-specific models. Specialized models are designed to address specific, frequently occurring reasons for video rejection. These models are operated on temporally subsampled frames to enable efficient processing.

1. *Cartoons and presenting*. This uses a classifier [15] to remove criteria in (C1) *e.g.* cartoon or screen recording; and (C3) *e.g.* static scenes of person sitting in front of camera or unlikely to interact.

2. *Non-perspective distortion*. Videos with high predicted distortion are removed. Distortion complicates applications that require reliable correspondences, so we consider them low quality (C1).
3. *Focal length*. Videos with high predicted focal length variance are removed, as they often contain zoom effects or shot changes (C2). We also exclude videos with long focal lengths, which typically include backgrounds that are too blurred for reliable pose estimation (C2) or are sporting shots with static cameras (C3).
4. *Dynamic object masking*. Videos with excessively large predicted mask sizes are considered unsuitable for pose estimation (C2) and are removed, as reliable pose estimation requires sufficient static correspondences.
5. *Optical flow*. High peak sequential optical flow is typically due to shot change (C2); such videos are removed using predicted flow. Low mean sequential flow typically indicates static videos (C3); these are also removed.
6. *Point tracking*. Abrupt disappearance of predicted point tracks can indicate shot change, blurred background, or severe zoom-in/zoom-out, making the video unsuitable for pose estimation (C2). Conversely, extremely stable tracks suggest neither the camera nor the scene has moved significantly (C3). Both cases are removed.

Filtering using general VLM. The generalist VLM addresses a variety of possibilities for video rejection. We prompt GPT-4o mini [55, 56] a series of eight questions addressing all three classes of criteria. They cover: if the video is taken from an ambiguous reference, if the background is too blurry to match correspondence, if frames are distorted or have very long focal length, if the scene is static, if the video is a cartoon, has been post-processed, and if children are present. We find having overlap with specialist filters, *e.g*. identifying long focal length, is helpful, since the VLM can be prompted to consider diverse cues for removal.

Collection details. Beginning from 3.2M Panda-70M [11] videos, we apply filtering (§ 3.2) leaving 137K dynamic videos. We estimate pose on 107K and drop trajectories with less than 80% of frames registered, leaving 100k videos. We apply filters sequentially for efficiency. First are lightweight Hands23, optical flow and focal length filters on the full set. We remove low scoring videos, leaving 1.63M videos. We next run the distort filter and remove low scores, leaving 1.53M; then tracking, leaving 679K; then masking, leaving 462K; and finally the VLM, after which we apply a strict final filter to get 137K for pose estimation.

3.3. Dynamic Camera Pose Estimation

Having collected suitable videos, our objective is to obtain high-quality camera poses. Pose estimation on dynamic Internet video is challenging: not only do dynamic objects occlude the underlying static scene, the static scene can change appearance making correspondence estimation difficult. Our approach addresses both challenges (Figure 3). For masking, we use state-of-the-art methods in semantics, interaction, motion, and tracking, with each component improving overall performance. For correspondences, we use the latest point tracking method [19] applied within a sliding window on the video. Finally, we apply global bundle adjustment [71], which has shown to be effective in handling the challenges posed by Internet videos [98].

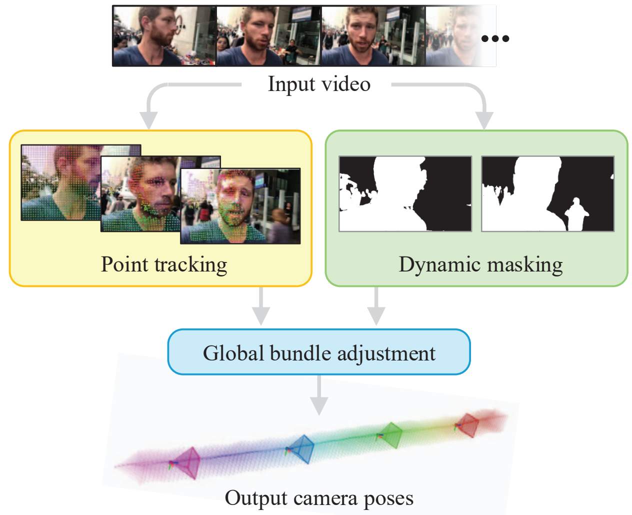

Figure 3. **Pose estimation approach**. We apply the state-of-the-art point tracking method at a sliding window to produce dense, long-term correspondences. Complementary dynamic masks are used to remove non-static tracks. The remaining static tracks are provided as input to global bundle adjustment.

Dynamic masking. We first aim to segment *dynamic* regions in an input video. Our approach uses advances in several communities to produce accurate masks.

1. *Semantic segmentation*. We segment common dynamic classes (*e.g*. humans, vehicles, animals, and sports equipment) using OneFormer [33].
2. *Object interaction segmentation*. This masks held objects which may dynamic yet outside semantic classes *e.g*. silverware. We use the Hands23 [15] hand-object interaction segmentation model.
3. *Motion segmentation*. This handles dynamic objects that are not common classes or involve human manipulation, *e.g*. rustling leaves or flowing water. We use motion masking of RoDynRF [47], which removes regions with high Sampson error [28] based on optical flow [73].
4. *Mask propagation*. Propagation provides smooth masks across frames and precise object boundaries. We use SAM2 [58] to propagate masks.

Point tracking. We estimate correspondences using point tracking, which uses temporal information in videos to improve estimation. Unlike pairwise estimation, tracking takes advantage of the fact that points generally move only small amounts between sequential frames. Additionally, tracking the same point over multiple frames yields rapidly increasing paired correspondences across each frame combination. For this purpose, we use BootsTAP [19] to track a grid of points forward several frames, move forward, and repeat in sliding window fashion. Grid-based point tracking facilitates denser correspondences, while the extended tracking duration supports long-term correspondences, reducing drift and aiding in loop closure. Sliding window tracking of point grids ensures that each frame maintains a sufficient number of correspondences in the case previous windows have become occluded or faced drift.

Global bundle adjustment. We use global bundle adjustment method Theia-SfM [71], using correspondences from tracklets as input. From a single tracklet, we extract correspondences for all pairs, excluding pairs containing a frame in which the tracklet is contained within a dynamic mask.

4. DynPose-100K: Dataset Analysis

In this work, we focus on dynamic Internet videos due to their large scale and diversity. We evaluate the effectiveness of our filtering pipeline in identifying videos suitable for pose estimation (§ 4.1) and provide a statistical analysis of the resulting dataset (§ 4.2).

4.1. Filtering Evaluation on Panda-Test

Our goal for filtering is to identify suitable videos for pose estimation given diverse Internet video.

Dataset. We evaluate filtering performance on 1,000 randomly selected held-out videos from Panda-70M, which we refer to as Panda-Test. Each video is manually classified as suitable or unsuitable for pose estimation, yielding 90 (9%) suitable videos. These 90 videos exhibit similar diversity to DynPose-100K, *e.g*. content, lengths, and dynamic sizes.

Metrics. Our primary evaluation metrics are precision and recall. We prioritize precision, aiming to minimize the number of inaccurate videos included in DynPose-100K. However, recall is also important, as a low recall would mean a larger number of necessary videos to process to achieve a dataset of the same size.

Baselines and ablations. We compare our method with existing and alternative filtering methods. We filter using Our SfM reconstructed points (CamCo [88]) and reprojection error (inspired by B-Timer [42]), along with video suitability and interaction classifier Hands23 [15]. In addition, we prompt GPT-4o mini [56] in a manner similar to our filtering pipeline (§ 3.2). We implement two versions of this comparison: one producing binary outputs and another assigning scores to each example. Finally, we conduct ablation studies on each step of our filtering method from § 3.2.

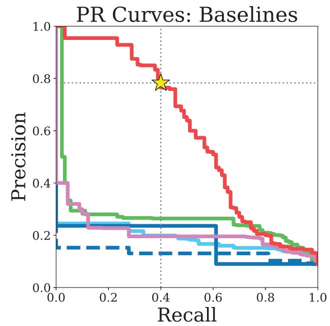

Figure 5. **Dynamic video filtering on Panda-Test.** We show PR curves for baselines and ablations. Our filtering surpasses all baselines and ablations by a considerable margin. The ☆ represents DynPose-100K's operating thresholds. For baselines, we show: ■ Reconstructed points (CamCo [88]), ■ Reprojection error, (solid ■) GPT-4o mini [56]: binary, (dashed ■) GPT-4o mini [56]: score, ■ Hands23 [15], and ■ Ours. For ablations, we begin from ■ Hands23 and add components until we recover ■ Ours. Specifically, we depict: ■ Hands23, ■ +Flow, ■ +Tracking, ■ +Masking, ■ +Focal, ■ +Distort, ■ +VLM (Ours).

Results. Figure 5 shows our filtering selects videos with high precision and recall, outperforming baseline methods. In fact, filtering has test precision of 0.78 at the threshold used for collecting DynPose-100K, a level that no baseline achieves at any recall, except for reprojection error at 0.02 recall. Each component in our method improves performance, with the VLM providing a large boost, even though it is relatively weak as a standalone approach.

4.2. Dataset Statistics

Having collected a substantial number of target videos, we assess the characteristics of the resulting dataset. In addition to evaluating the dataset size, we examine whether resulting videos exhibit desirable attributes, *e.g*. diverse content, appropriate video length and diverse dynamic object size.

Dataset size. Table 1 shows that DynPose-100K contains many more videos than existing diverse datasets. The largest alternatives are typically limited to specific settings, such as kitchens, driving, walking, or pet turntable videos [25, 34, 67, 70, 76]. Concurrent works CamCo [88] and B-Timer [42] also feature diverse content but have fewer videos and are private.

Video content. Figure 4 shows frequent nouns and verbs that appear in Panda-70M's [11] captions associated with DynPose-100K videos. Common nouns cover a wide range of subjects, objects and settings; while verbs reflect a variety of actions. This linguistic variety provides evidence DynPose-100K contains videos with diverse content.

Dataset	Real/Syn.	Num. vids.	Num. frames	Domain	Access
T.Air Shibuya [57]	Syn.	7	0.7K	Street	**Public**
MPI Sintel [8]	Syn.	14	0.7K	Scripted	**Public**
PointOdyssey [100]	Syn.	131	200K	Walking	**Public**
FlyingThings3D [48]	Syn.	220	2K	Objects	**Public**
Kubric Movi-E [26]	Syn.	400	10K	Objects	**Public**
EpicFields [76]	Real	671	19,000K	Kitchens	**Public**
Waymo [70]	Real	1,150	200K	Driving	**Public**
CoP3D [67]	Real	4,200	600K	Pets	**Public**
Ego-Exo4D [25]	Real	5,035	**23,000K**	Set tasks	**Public**
Stereo4D [34]	Real	**110,000**	10,000K	S. fisheye	**Public**
CamCo [88]	Real	12,000	385K	**Diverse**	Private
B-Timer [42]	Real	40,000	19,000K	**Diverse**	Private
DynPose-100K	Real	100,131	6,806K	**Diverse**	**Public**

Table 1. **Dynamic camera pose datasets.** DynPose-100K has the most videos of diverse Internet video datasets. Datasets with more frames are private or more uniform; *e.g.* stereo fisheye is typically outdoor PoV walking. We use short videos, yielding fewer frames but high dynamics (Figure 6).

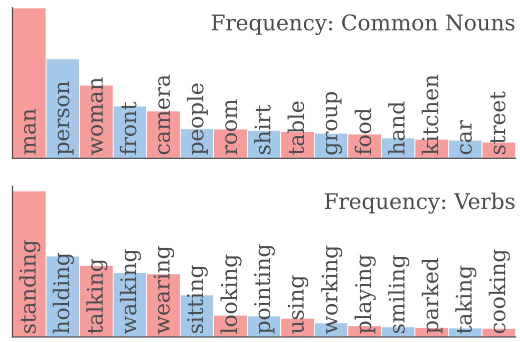

Figure 4. **Diverse content.** Frequent nouns cover varied subjects: person, hand, car; objects: shirt, table, food; and settings: room, kitchen, street. Verbs span diverse actions: using, working, playing.

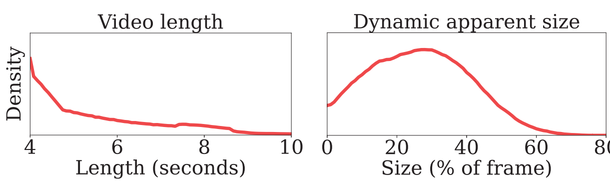

Figure 6. **Left: Targeted video length.** DynPose-100K videos are primarily 4-10s, ideal for dynamic pose: shorter videos contain little ego-motion, longer videos have less dense dynamics and ego-motion. **Right: Diverse dynamic apparent size.** Mean size in % across video. Large dynamic objects occlude static correspondences, making pose estimation challenging. Videos may average small size in the case of only a few dynamic frames.

Video length. Figure 6, left shows the distribution of video lengths in DynPose-100K. The majority of videos span 4 to 10 seconds. These short clips are typically rich in dynamics, while being long enough for substantial camera motion.

Dynamic object size. Figure 6, right shows the distribution of apparent dynamic object sizes in DynPose-100K, ranging from small to large. Large dynamic objects occlude static correspondences, making pose estimation challenging. Smaller apparent dynamic objects typically correspond to objects further away, which can move quickly, making precise masking challenging. Videos where dynamic objects occupy nearly the entire frame are filtered out as they make pose estimation infeasible.

5. Evaluation on Camera Pose Estimation

We proceed to evaluate the pose estimation efficacy of the curated dataset DynPose-100K through controlled experiments. Direct evaluation on dynamic Internet videos is challenging due to the lack of ground truth camera pose data. Our evaluation approach is therefore two-fold: (1) we use a photorealistic synthetic rendering dataset, Lightspeed, which provides ground truth poses for direct comparison; (2) we evaluate directly on Internet videos by annotating 10K precise correspondences on Panda-Test, enabling pose evaluation using Sampson error [28].

Baselines and ablations. Our primary point of comparison throughout is ParticleSfM [98], which shares our general pipeline involving static tracking, dynamic masking, and global SfM.We also compare against standard SfM method COLMAP [61], both in its original form and with correspondences filtered using dynamic masking. Finally, we evaluate our method against state-of-the-art learning-based static methods DROID-SLAM [74] and DUSt3R [81], and dynamic methods LEAP-VO [12] and MonST3R [95].

5.1. Pose Evaluation on Lightspeed

We introduce Lightspeed, a challenging, photorealistic benchmark for dynamic pose estimation containing ground truth camera pose. This dataset is a good benchmark for pose estimation because it is shares several important characteristics of DynPose-100K: diverse environments, large dynamic objects and varied clip lengths of several seconds.

Dataset. We use NVIDIA Racer RTX [53], shown in Figure 7, which follows several RC cars moving quickly through challenging indoor and outdoor scenes, featuring day and night lighting and first and third person views. We split the video into clips, removing ones with non-perspective distortion, blurry background, static scenes, or cameras with static or have trivial linear trajectory (following [87]), resulting in 36 sequences. We extract ground truth camera pose from corresponding 3D Assets.

Metrics. We follow prior work [47, 64, 98] in reporting Average Trajectory Error (ATE) as well as Relative Pose Error (RPE) for rotation and translation. For full details see prior

	All 36 videos (Iden. Rot. + Rand. Tr. if fail)				8 videos: all succeed only		
Method	% Vids. reg.↑	ATE (m)↓	RPE Tr. (m)↓	RPE Rot. (°)↓	ATE (m)↓	RPE Tr. (m)↓	RPE Rot. (°)↓
Iden. Rot. + Rand. Tr.	**100.**	0.652	0.139	1.60	0.390	0.080	1.00
DROID-SLAM [74]	**100.**	0.198	0.046	1.75	0.048	0.017	0.82
DUSt3R [81]	97.2	0.412	0.177	20.1	0.256	0.124	18.5
MonST3R [95]	**100.**	0.149	0.046	**1.21**	0.036	0.011	0.46
LEAP-VO [12]	**100.**	0.206	0.049	1.70	0.037	0.011	0.73
COLMAP [61]	44.4	0.388	0.082	2.03	0.122	0.026	1.91
COLMAP+Mask [61]	38.9	0.323	0.085	1.64	0.089	0.017	1.36
ParticleSfM [98]	97.2	0.185	0.075	2.99	0.051	0.047	2.91
Ours	**100.**	**0.072**	**0.033**	1.31	**0.003**	**0.002**	**0.30**

Table 2. **Camera pose estimation on Lightspeed.** Our pose estimation algorithm registers all sequences and cuts trajectory error by 50% across all sequences (left) and 90% on easy sequences (right) *vs.* all other methods.

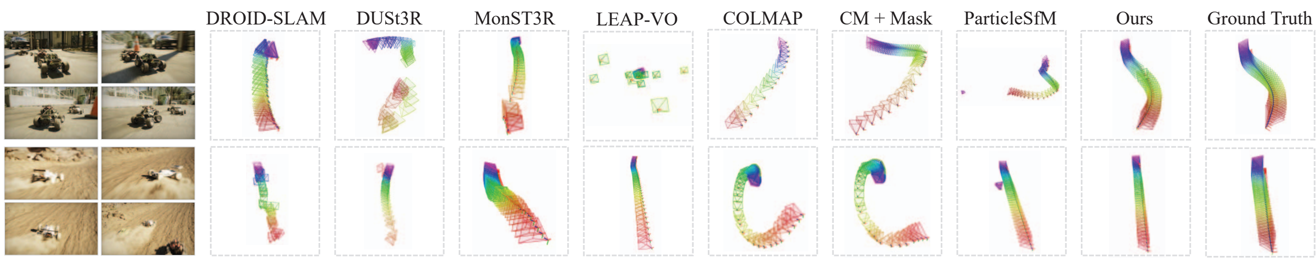

Figure 7. **Predicted trajectories on Lightspeed**. Pose sequence over time: ROYGBV. We visualize photorealistic renderings of Lightspeed left. Static methods DROID-SLAM, COLMAP and DUSt3R struggle in this dynamic setting, failing to register a consistent sequence or putting too much or too little curvature. Top: dynamic methods MonST3R, LEAP-VO and ParticleSfM do not produce smooth sequences. Bottom: MonST3R, LEAP-VO and COLMAP+Mask add curvature. Ours produces smooth and accurate trajectories.

work. To summarize, metrics are computed after trajectory alignment and scaling, and relative error is computed on sequential frames. To compute scores across all videos, we replace predicted trajectories that do not converge with random uniform translations and identity rotations. We also separately report scores on the subset of videos upon which all methods converge.

Results. Table 2 shows COLMAP and COLMAP+Mask struggle to register many challenging sequences in Lightspeed. DROID-SLAM, DUSt3R, LEAP-VO and ParticleSfM provide registration, but are inaccurate. MonST3R offers better trajectory error, though Ours is clearly superior, cutting trajectory error by 50% (all videos) and 90% (succeed only). All alternatives struggle in at least one sequence in Figure 7, while Ours handles both.

5.2. Pose Evaluation on Panda-Test

Next, we evaluate on dynamic Internet videos. We use the 90-video subset from Panda-Test (§ 4.1), featuring various challenging objects with estimable camera poses.

Metrics. Ground truth camera poses are unavailable in unlabeled dynamic Internet videos, so we opt to annotate correspondences and compute Sampson error from the predicted poses. We curate 10K correspondences across unique image pairs spanning the 90 videos to use for evaluation. Pairs are manually annotated by expert annotators to achieve coarse accuracy. These matches are refined using SuperPoint [17] and LightGlue [44] by selecting the LightGlue match found within 10 pixels of the human-labeled points on 720p images. If no such match is found, the correspondence is discarded. Frame pairs are selected randomly from within 2.5 seconds of each other.

Reprojection error for each pair is the square root of Sampson error. Sampson error computes the squared distance from an annotated correspondence to the epipolar line computed using the fundamental matrix based on predicted relative camera pose and intrinsics. Metrics are averaged for all pairs in each video, enabling per-video analysis. Our main metrics are mean error across videos, and percent of videos with mean reprojection error within a threshold; we use 5, 10 and 30 pixels at normalized 720p resolution. Non-registered frames are filled with nearest neighbor poses. If an entire sequence is not registered, the identity is used.

Results. Table 3 shows similar trends to Lightspeed: static methods DROID-SLAM and DUSt3R are inaccurate while COLMAP, even with dynamic masking, struggles to register many videos. ParticleSfM improves, but contains high mean error. MonST3R and LEAP-VO register all frames, but Ours reduces error across metrics. This in-

Baselines	% Vids. registered↑	All 90 estimable videos (Identity if fail) Mean per-video reprojection error, 720p % < 5Pix↑	% < 10 Pix↑	% < 30 Pix↑	Mean↓	52 videos: all succeed only Mean per-video reprojection error, 720p % < 5 Pix↑	% < 10 Pix↑	% < 30 Pix↑	Mean↓
Identity	**100.**	0.0	0.0	2.2	151	0.0	0.0	0.0	133
DROID-SLAM [74]	**100.**	57.8	77.8	94.4	11.0	59.6	84.6	96.2	6.78
DUSt3R [81]	96.7	0.0	6.7	48.9	43.0	0.0	9.6	57.7	30.3
MonST3R [95]	**100.**	55.6	78.9	90.0	9.86	63.5	84.6	90.4	9.71
LEAP-VO [12]	**100.**	64.4	76.7	96.7	7.59	75.0	84.6	98.1	6.03
COLMAP [61]	82.2	51.1	62.2	73.3	27.5	71.2	82.7	92.3	9.03
COLMAP+Mask [61]	67.8	47.8	58.9	75.6	30.1	69.2	82.7	96.2	6.10
ParticleSfM [98]	92.2	70.0	76.7	88.9	12.5	80.8	86.5	96.2	6.77
Ours	95.6	**72.2**	**84.4**	**98.9**	**5.76**	**82.7**	**94.2**	**100.**	**3.75**

Table 3. **Camera pose estimation on Panda-Test.** Reprojection error on 10K image pairs, by video, normalized to 720p. Static methods DUSt3R and COLMAP struggle faced with dynamics while DROID-SLAM lacks precision. ParticleSfM registers more videos than COLMAP+Mask but both fall short of Ours in registration and accuracy. MonST3R and LEAP-VO register all frames leading to moderate errors, but Ours outperforms across metrics.

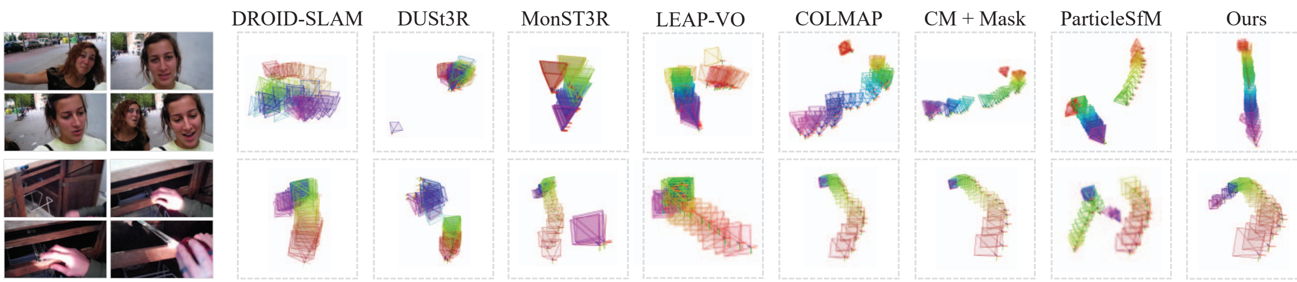

Figure 8. **Predicted trajectories on Panda-Test**. Pose sequence over time: ROYGBV. Static methods like DROID-SLAM, DUSt3R and COLMAP struggle faced with dynamics. MonST3R, LEAP-VO, COLMAP+Mask and ParticleSfM can struggle with large dynamic regions (top) and tracking across varied appearance and lighting (bottom); while Ours handles these cases.

Method	% Vids. reg.↑	All 90 estimable videos (Identity if fail) Mean per-video reprojection error, 720p % < 5Px↑	% < 10 Px↑	% < 30 Px↑	Mean↓
DUSt3R	96.7	0.0	6.7	48.9	43.0
+ Synthetic (MonST3R [95])	**100.**	**55.6**	78.9	90.0	9.86
+ DynPose-100K (Ours)	**100.**	54.4	**82.2**	**92.2**	**8.78**

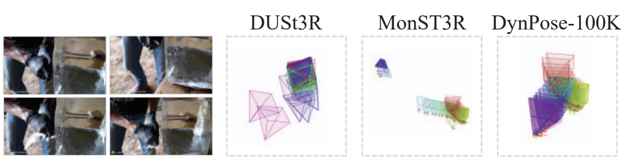

Table 4. **Camera estimation on Panda-Test.** DynPose-100K fine-tuning of DUSt3R has lower mean error and similar to or better than accuracy compared to synthetic data (MonST3R). We train with only 2K videos / 140K frames, smaller than the 1.3M frames used to train MonST3R; demonstrating efficient supervision.

cludes more than 35% reduction in mean error vs. all methods on the 52 sequences all methods succeed. Figure 8 shows qualitative results from two challenging sequences. Qualitative results agree with quantitative: DROID-SLAM, DUSt3R and COLMAP struggle, while MonST3R, LEAP-VO, COLMAP+Mask and ParticleSfM face failure cases. Ours better handles difficult cases *e.g*. large dynamics or varied appearance. Evidently, the combined state-of-the-art components used for masking and tracking (§ 3.3) yield a pipeline effective at each corresponding task.

6. Conclusion

In this paper, we introduce DynPose-100K, a large-scale dataset of dynamic Internet video annotated with camera poses. It is curated using carefully designed video filtering and camera pose estimation pipelines, validated through experiments. We hope DynPose-100K will enable exciting new possibilities. For instance, Table 4 shows the dataset can fine-tune DUSt3R [81] to produce lower mean error than the synthetic data of MonST3R [95]. To supplement DynPose-100K, we collect high-quality synthetic dataset Lightspeed, enabling ground truth pose benchmarking.

Acknowledgments. We thank Gabriele Leone and the NVIDIA Lightspeed Content Tech team for sharing the original 3D assets and scene data for creating the Lightspeed benchmark. We thank Yunhao Ge, Zekun Hao, Yin Cui, Xiaohui Zeng, Zhaoshuo Li, Hanzi Mao, Jiahui Huang, Justin Johnson, JJ Park and Andrew Owens for invaluable inspirations, discussions and feedback on this project.

References

[1] Shikhar Bahl, Abhinav Gupta, and Deepak Pathak. Human-to-robot imitation in the wild. In *RSS*, 2022. 2

[2] Sherwin Bahmani, Xian Liu, Yifan Wang, Ivan Skorokhodov, Victor Rong, Ziwei Liu, Xihui Liu, Jeong Joon Park, Sergey Tulyakov, Gordon Wetzstein, et al. Tc4d: Trajectory-conditioned text-to-4d generation. In *ECCV*, 2024. 2

[3] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. 2

[4] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Yuri Feigin, Peter Fu, Thomas Gebauer, Daniel Kurz, Tal Dimry, Brandon Joffe, Arik Schwartz, et al. Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. In *NeurIPS*, 2021. 2

[5] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv*, 2023. 2

[6] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 2

[7] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators, 2024. 2

[8] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012. 2, 6

[9] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *CVPR*, 2022. 2

[10] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. In *3DV*, 2017. 2

[11] Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, et al. Panda-70m: Captioning 70m videos with multiple cross-modality teachers. In *CVPR*, 2024. 2, 3, 4, 5

[12] Weirong Chen, Le Chen, Rui Wang, and Marc Pollefeys. Leap-vo: Long-term effective any point tracking for visual odometry. In *CVPR*, 2024. 2, 6, 7, 8

[13] Zilong Chen, Yikai Wang, Feng Wang, Zhengyi Wang, and Huaping Liu. V3d: Video diffusion models are effective 3d generators. *arXiv*, 2024. 2

[14] Ho Kei Cheng, Seoung Wug Oh, Brian Price, Joon-Young Lee, and Alexander Schwing. Putting the object back into video object segmentation. In *CVPR*, 2024. 2

[15] Tianyi Cheng, Dandan Shan, Ayda Hassen, Richard Higgins, and David Fouhey. Towards a richer 2d understanding of hands at scale. *NeurIPS*, 2023. 3, 4, 5

[16] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. In *NeurIPS*, 2024. 2

[17] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPRW*, 2018. 7

[18] Carl Doersch, Yi Yang, Mel Vecerik, Dilara Gokay, Ankush Gupta, Yusuf Aytar, Joao Carreira, and Andrew Zisserman. Tapir: Tracking any point with per-frame initialization and temporal refinement. In *ICCV*, 2023. 3

[19] Carl Doersch, Yi Yang, Dilara Gokay, Pauline Luc, Skanda Koppula, Ankush Gupta, Joseph Heyward, Ross Goroshin, João Carreira, and Andrew Zisserman. Bootstap: Bootstrapped training for tracking-any-point. In *ACCV*, 2024. 3, 4, 5

[20] Bardienus Duisterhof, Lojze Zust, Philippe Weinzaepfel, Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Mast3r-sfm: a fully-integrated solution for unconstrained structure-from-motion. *arXiv*, 2024. 2

[21] Zipeng Fu, Tony Z Zhao, and Chelsea Finn. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. In *CoRL*, 2024. 2

[22] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. Get3d: A generative model of high quality 3d textured shapes learned from images. In *NeurIPS*, 2022. 2

[23] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. In *ICLR*, 2024. 2

[24] Lily Goli, Sara Sabour, Mark Matthews, Marcus Brubaker, Dmitry Lagun, Alec Jacobson, David J Fleet, Saurabh Saxena, and Andrea Tagliasacchi. Romo: Robust motion segmentation improves structure from motion. *arXiv*, 2024. 2

[25] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In *CVPR*, 2024. 2, 5, 6

[26] 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 *CVPR*, 2022. 2, 6

[27] Adam W Harley, Zhaoyuan Fang, and Katerina Fragkiadaki. Particle video revisited: Tracking through occlusions using point trajectories. In *ECCV*, 2022. 3

[28] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 2, 4, 6

[29] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for text-to-video generation. *arXiv*, 2024. 2

[30] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 2
[31] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv*, 2022. 2
[32] 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*, 2023. 2
[33] Jitesh Jain, Jiachen Li, Mang Tik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. Oneformer: One transformer to rule universal image segmentation. In *CVPR*, 2023. 2, 4
[34] Linyi Jin, Richard Tucker, Zhengqi Li, David Fouhey, Noah Snavely, and Aleksander Holynski. Stereo4d: Learning how things move in 3d from internet stereo videos. In *CVPR*, 2025. 2, 5, 6
[35] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. In *ECCV*, 2024. 3
[36] Yoni Kasten, Wuyue Lu, and Haggai Maron. Fast encoder-based 3d from casual videos via point track processing. In *NeurIPS*, 2024. 3
[37] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. In *SIGGRAPH*, 2017. 2
[38] Johannes Kopf, Xuejian Rong, and Jia-Bin Huang. Robust consistent video depth estimation. In *CVPR*, 2021. 2
[39] Zhengfei Kuang, Shengqu Cai, Hao He, Yinghao Xu, Hongsheng Li, Leonidas Guibas, and Gordon Wetzstein. Collaborative video diffusion: Consistent multi-video generation with camera control. In *NeurIPS*, 2024. 2
[40] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *CVPR*, 2021. 2
[41] 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. In *CVPR*, 2025. 3
[42] Hanxue Liang, Jiawei Ren, Ashkan Mirzaei, Antonio Torralba, Ziwei Liu, Igor Gilitschenski, Sanja Fidler, Cengiz Oztireli, Huan Ling, Zan Gojcic, and Jiahui Huang. Feed-forward bullet-time reconstruction of dynamic scenes from monocular videos. *arXiv*, 2024. 2, 5, 6
[43] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 3
[44] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. LightGlue: Local Feature Matching at Light Speed. In *ICCV*, 2023. 7
[45] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *CVPR*, 2024. 2
[46] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *ICCV*, 2023. 2
[47] Yu-Lun Liu, Chen Gao, Andreas Meuleman, Hung-Yu Tseng, Ayush Saraf, Changil Kim, Yung-Yu Chuang, Johannes Kopf, and Jia-Bin Huang. Robust dynamic radiance fields. In *CVPR*, 2023. 2, 4, 6
[48] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016. 2, 3, 6
[49] 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 *CVPR*, 2023. 2
[50] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. ORB-SLAM: a versatile and accurate monocular SLAM system. *T-RO*, 2015. 2
[51] Haomiao Ni, Changhao Shi, Kai Li, Sharon X Huang, and Martin Renqiang Min. Conditional image-to-video generation with latent flow diffusion models. In *CVPR*, 2023. 2
[52] NVIDIA. Cosmos world foundation model platform for physical ai. *arXiv*, 2025. 2
[53] NVIDIA GeForce. NVIDIA Racer RTX | The future of graphics powered by GeForce RTX 40 Series, 2022. [Online; accessed Access Date: 2024-11-9]. 6
[54] Abby O'Neill, Abdul Rehman, Abhinav Gupta, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, et al. Open x-embodiment: Robotic learning datasets and rt-x models. In *CoRL*, 2023. 2
[55] OpenAI. Gpt-4 technical report, 2023. 4
[56] OpenAI. Gpt-4o mini: advancing cost-efficient intelligence, 2024. 3, 4, 5
[57] Yuheng Qiu, Chen Wang, Wenshan Wang, Mina Henein, and Sebastian Scherer. Airdos: Dynamic slam benefits from articulated objects. In *ICRA*, 2022. 2, 6
[58] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv*, 2024. 2, 3, 4
[59] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *ICCV*, 2021. 2
[60] Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas MÃ¼ller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *CVPR*, 2025. 2
[61] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 2, 6, 7, 8

[62] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*, 2017. 2

[63] Jenny Seidenschwarz, Qunjie Zhou, Bardienus Duisterhof, Deva Ramanan, and Laura Leal-Taixé. Dynomo: Online point tracking by dynamic online monocular gaussian reconstruction. *arXiv*, 2024. 3

[64] Shihao Shen, Yilin Cai, Wenshan Wang, and Sebastian Scherer. Dytanvo: Joint refinement of visual odometry and motion segmentation in dynamic environments. In *ICRA*, 2023. 2, 6

[65] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv*, 2022. 2

[66] Uriel Singer, Shelly Sheynin, Adam Polyak, Oron Ashual, Iurii Makarov, Filippos Kokkinos, Naman Goyal, Andrea Vedaldi, Devi Parikh, Justin Johnson, et al. Text-to-4d dynamic scene generation. *arXiv*, 2023. 2

[67] Samarth Sinha, Roman Shapovalov, Jeremy Reizenstein, Ignacio Rocco, Natalia Neverova, Andrea Vedaldi, and David Novotny. Common pets in 3d: Dynamic new-view synthesis of real-life deformable categories. In *CVPR*, 2023. 2, 3, 5, 6

[68] Cameron Smith, David Charatan, Ayush Tewari, and Vincent Sitzmann. Flowmap: High-quality camera poses, intrinsics, and depth via gradient descent. In *3DV*, 2025. 2

[69] Noah Snavely. Scene reconstruction and visualization from internet photo collections. *PhD Thesis*, 2008. 2

[70] 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 *CVPR*, 2020. 2, 5, 6

[71] Chris Sweeney. Theia multiview geometry library: Tutorial & reference. `http://theia-sfm.org`. 2, 3, 4, 5

[72] Zachary Teed and Jia Deng. Deepv2d: Video to depth with differentiable structure from motion. *arXiv*, 2018. 2

[73] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020. 2, 4

[74] Zachary Teed and Jia Deng. DROID-SLAM: Deep visual SLAM for monocular, stereo, and RGB-D cameras. *NeurIPS*, 2021. 3, 6, 7, 8

[75] Zachary Teed, Lahav Lipson, and Jia Deng. Deep patch visual odometry. In *NeurIPS*, 2023. 2

[76] Vadim Tschernezki, Ahmad Darkhalil, Zhifan Zhu, David Fouhey, Iro Laina, Diane Larlus, Dima Damen, and Andrea Vedaldi. Epic fields: Marrying 3d geometry and video understanding. *NeurIPS*, 2024. 2, 5, 6

[77] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. Vggsfm: Visual geometry grounded deep structure from motion. In *CVPR*, 2024. 2

[78] 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 *CVPR*, 2021. 2

[79] Qianqian Wang, Yen-Yu Chang, Ruojin Cai, Zhengqi Li, Bharath Hariharan, Aleksander Holynski, and Noah Snavely. Tracking everything everywhere all at once. In *ICCV*, 2023. 3

[80] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *CVPR*, 2025. 2

[81] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024. 3, 6, 7, 8

[82] 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 *IROS*, 2020. 2

[83] Wenshan Wang, Yaoyu Hu, and Sebastian Scherer. TartanVO: A generalizable learning-based VO. In *CoRL*, 2021. 2

[84] Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, and Jiwen Lu. Drivedreamer: Towards real-world-driven world models for autonomous driving. In *ECCV*, 2024. 2

[85] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation. In *SIGGRAPH*, 2024. 2

[86] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. Synsin: End-to-end view synthesis from a single image. In *CVPR*, 2020. 2

[87] J. Wulff, D. J. Butler, G. B. Stanley, and M. J. Black. Lessons and insights from creating a synthetic optical flow benchmark. In *ECCVW*, 2012. 6

[88] Dejia Xu, Weili Nie, Chao Liu, Sifei Liu, Jan Kautz, Zhangyang Wang, and Arash Vahdat. Camco: Camera-controllable 3d-consistent image-to-video generation. *arXiv*, 2024. 2, 5, 6

[89] Hongwei Xue, Tiankai Hang, Yanhong Zeng, Yuchong Sun, Bei Liu, Huan Yang, Jianlong Fu, and Baining Guo. Advancing high-resolution video-language representation with large-scale video transcriptions. In *CVPR*, 2022. 2

[90] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation. In *SIGGRAPH Asia*, 2023. 2

[91] Weicai Ye, Xinyu Chen, Ruohao Zhan, Di Huang, Xiaoshui Huang, Haoyi Zhu, Hujun Bao, Wanli Ouyang, Tong He, and Guofeng Zhang. Datap-sfm: Dynamic-aware tracking any point for robust dense structure from motion in the wild. *arxiv*, 2024. 3

[92] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *ICCV*, 2023. 2

[93] Zhichao Yin and Jianping Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *CVPR*, 2018. 2

[94] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *CVPR*, 2021. 2

[95] 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 *ICLR*, 2025. 3, 6, 7, 8

[96] Zhoutong Zhang, Forrester Cole, Zhengqi Li, Michael Rubinstein, Noah Snavely, and William T Freeman. Structure and motion from casual videos. In *ECCV*, 2022. 2

[97] Zhenghao Zhang, Junchao Liao, Menghao Li, Zuozhuo Dai, Bingxue Qiu, Siyu Zhu, Long Qin, and Weizhi Wang. Tora: Trajectory-oriented diffusion transformer for video generation. *arXiv*, 2024. 2

[98] Wang Zhao, Shaohui Liu, Hengkai Guo, Wenping Wang, and Yong-Jin Liu. Particlesfm: Exploiting dense point trajectories for localizing moving cameras in the wild. In *ECCV*, 2022. 2, 4, 6, 7, 8

[99] Yuyang Zhao, Chung-Ching Lin, Kevin Lin, Zhiwen Yan, Linjie Li, Zhengyuan Yang, Jianfeng Wang, Gim Hee Lee, and Lijuan Wang. Genxd: Generating any 3d and 4d scenes. In *ICLR*, 2025. 2

[100] 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

[101] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. In *SIGGRAPH*, 2018. 2