

AnyView: Synthesizing Any Novel View in Dynamic Scenes

Basile Van Hoorick¹ Dian Chen¹ Shun Iwase¹ Pavel Tokmakov¹
Muhammad Zubair Irshad¹ Igor Vasiljevic¹ Swati Gupta¹ Fangzhou Cheng^{1,2}
Sergey Zakharov¹ Vitor Campagnolo Guizilini¹

¹Toyota Research Institute ²Amazon Web Services

tri-ml.github.io/AnyView

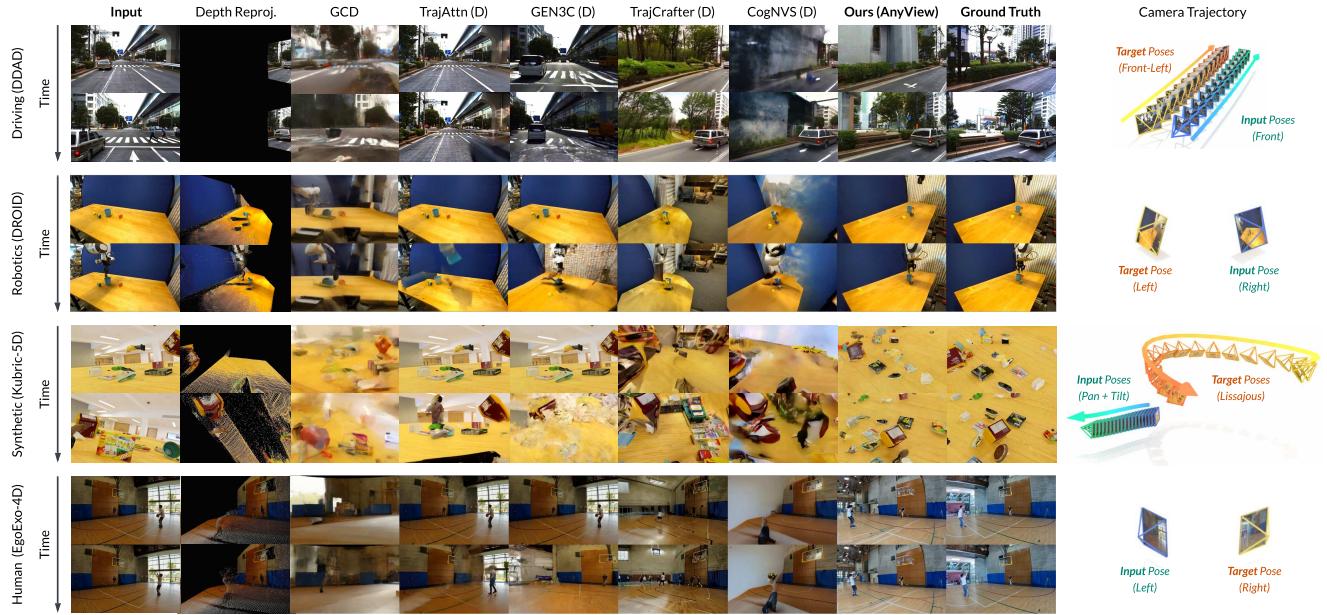


Figure 1. **Enabling consistent extreme monocular dynamic view synthesis:** We introduce *AnyView*, a diffusion framework that can generate videos of dynamic scenes from *any* chosen perspective, conditioned on a single input video. Our model operates end-to-end, without explicit scene reconstruction or expensive test-time optimization techniques. Existing methods tend to fail to extrapolate, largely copying the input view. More recent baselines can recover the overall structure in some cases (1st, 2nd rows), but fail when the camera trajectories become more complex (3rd row). Meanwhile, our method preserves scene geometry, appearance, and dynamics, despite working with drastically different target poses and highly “incomplete” visual observations. (D) indicates a baseline that relies on reprojected point clouds from estimated depth maps.

Abstract

Modern generative video models excel at producing convincing, high-quality outputs, but struggle to maintain multi-view and spatiotemporal consistency in highly dynamic real-world environments. In this work, we introduce *AnyView*, a diffusion-based video generation framework for dynamic view synthesis with minimal inductive biases or geometric assumptions. We leverage multiple data sources with various levels of supervision, including monocular (2D), multi-view static (3D) and multi-view dynamic (4D) datasets, to train a generalist spatiotempo-

ral implicit representation capable of producing zero-shot novel videos from arbitrary camera locations and trajectories. We evaluate *AnyView* on standard benchmarks, showing competitive results with the current state of the art, and propose *AnyViewBench*, a challenging new benchmark tailored towards extreme dynamic view synthesis in diverse real-world scenarios. In this more dramatic setting, we find that most baselines drastically degrade in performance, as they require significant overlap between viewpoints, while *AnyView* maintains the ability to produce realistic, plausible, and spatiotemporally consistent videos when prompted from any viewpoint.

1. Introduction

Generating a new video from an arbitrary camera perspective while the scene is in motion is a highly ambitious and fundamentally under-constrained task. A single input view only depicts a fraction of the world; the rest is occluded, transient, or simply unknown. New moving objects may enter the scene at any moment, and unobserved regions might be dynamic themselves, further introducing uncertainty into the generative process. Exact 4D reconstruction from such signals is therefore impractical in the general case. For many downstream uses of 4D video representations [20, 32, 55], — such as robotics, world models, simulation, telepresence, VR/AR, autonomous driving — what matters is not an exact correspondence with ground truth, but rather whether the resulting representation is realistic, temporally stable, and self-consistent across large viewpoint changes. A common problem with learned visuomotor policies, for example, is that they often suffer from brittleness under shifting camera poses [10, 39, 49, 62].

Humans routinely engage this problem in a way that is both rooted in intuition and very useful in practice: as we observe the physical world, we mentally “re-project” scenes, inferring likely layouts, object shapes, scene completions, and plausible dynamics from limited information [5, 8, 12, 28, 30, 35, 40, 46]. This is not simply a low-level reconstruction capability: it is a powerful prior over shapes, semantics, materials, and motion that yields predictions that are largely viewpoint-invariant. The goal of this paper is to take a step towards solving that objective: we target perceptually realistic 4D video synthesis under extreme camera trajectories and displacements. To that end, we endow video generative models with the same inductive bias: to produce reasonable scene completions, based on a single input video, that respect scene geometry, physics, and object permanence, even when there is little overlap with the conditioning view.

Most existing *dynamic view synthesis* (DVS) approaches and benchmarks are not built for this regime [13, 14, 25, 53, 57], as they typically operate in *narrow* settings: the input and target cameras are spatially nearby, looking in similar directions, and thus methods are designed to maximize pixel metrics under limited motion, ignoring the rest of the scene. In particular, most current state of the art DVS methods [9, 66, 68] rely on explicit 3D reconstructions (i.e., depth reprojection + image inpainting), costly test-time optimization and finetuning techniques, and support a limited set of camera trajectories.

To move away from this simplified setting, we first present **AnyView**, a novel diffusion-based DVS architecture for high-fidelity video-to-video synthesis under dramatic camera trajectory changes, capable of producing perceptually plausible and semantically consistent videos from arbitrary novel viewpoints. Our framework is purposefully light

on explicit inductive biases: camera parameters are provided via dense ray-space conditioning, allowing us to support any model (including non-pinhole), and the network learns to synthesize unobserved content implicitly, guided by large-scale, diverse training data. To reach this level of implicit 4D understanding, we leverage existing video foundation models as a source of rich internet-scale 2D appearance and motion priors, and augment them by incorporating multi-view geometry and camera controllability, learned using 12 multi-domain 3D and 4D datasets.

Secondly, due to the aforementioned shortcomings of existing evaluation procedures, we assemble **AnyViewBench**, a novel benchmark that formalizes and standardizes the *extreme* DVS task across various domains (driving, robotics, and human activity), camera rigs (ego-centric and exocentric), and camera motion patterns (fixed, linear, or complex, sometimes with changing intrinsics). Each scene provides at least two time-synchronized views, enabling rigorous metric evaluations with ground truth videos without resorting to proxy setups.

2. Related Work

2.1. Video Generative Models

In recent years, significant advances have been made in video generation, leading to the development of increasingly capable generative models. Stability AI’s SVD [7] pioneered video diffusion by adding temporal layers to a pre-trained image diffusion network [44], allowing coherent short video clip generation from single images or text prompts. CogVideoX [64] introduced a 3D Variational Autoencoder (VAE) to compress videos across spatial and temporal dimensions, enhancing both compression rate and video fidelity. NVIDIA’s Cosmos [2] introduced a suite of models with strong long-range temporal consistency and flexible conditioning signals (text, image and video input). Wan [52] is a novel mixture of experts-based video generation architecture, and provides a suite of video world models that excel at prompt following and photorealistic generation. However, none of these architectures were originally designed with camera conditioning in mind, focusing instead on future frame forecasting in the single – or more recently multi-camera [37] – setting.

2.2. Dynamic View Synthesis

Dynamic view synthesis is the task of generating novel renderings from arbitrary viewpoints and timesteps given a monocular video of a dynamic scene. A number of works have combined video generation with explicit geometric conditioning to improve geometric 3D consistency and control [21, 51, 58, 70]. Shape of Motion [53] addresses monocular dynamic reconstruction by representing scene motion through a compact set of SE(3) motion bases, en-

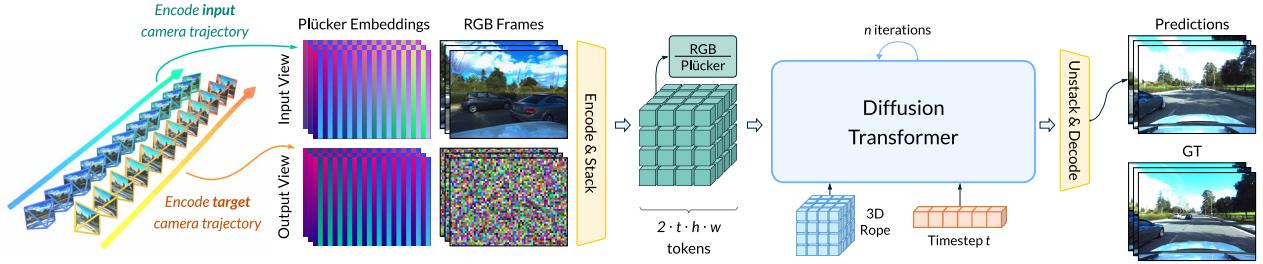


Figure 2. **The AnyView architecture.** For both the clean input and noisy target videos, we concatenate pixels (RGB values) and camera information (Plücker vectors) belonging to the *same* viewpoint along the *channel* dimension, after independently encoding each modality into latent embeddings. We then stack these two multimodal videos along the *sequence dimension*, for a total of $2 \cdot t \cdot h \cdot w$ tokens, which are fed into the diffusion transformer to iteratively denoise the target video.

abling soft segmentation into multiple rigidly moving parts using monocular depth and long-range 2D tracks. It fuses monocular depth and long-range 2D tracks to obtain a globally consistent dynamic 3D representation.

While explicit modeling approaches can achieve relatively high accuracy, they are computationally expensive and brittle. GCD [50] proposed to address dynamic view synthesis as an implicit problem, by re-purposing internet-scale video diffusion models via camera conditioning. This implicit formulation provides the greatest flexibility and robustness, but requires ground truth multi-view video data for training. ReCamMaster [4] advanced this research direction by utilizing a more powerful video generation model and a more realistic simulator to generate training data, whereas Trajectory Attention [60] augments video diffusion models with a trajectory-aware attention mechanism, improving fine-grained camera motion control and temporal consistency. AC3D [3] analyzes how video diffusion models internally represent 3D camera motion, adding ControlNet-style conditioning to improve controllability.

Other methods [9, 43, 68] have taken a hybrid approach by first lifting the input video in 3D via monocular depth estimation, reprojecting the resulting point cloud to the target camera pose, and then treating dynamic view synthesis as an in-painting problem. Among these methods, CogNVS [9] further introduces test-time optimization to improve rendering accuracy at the cost of inference speed, while StreetCrafter [61] focuses on autonomous driving scene generation, utilizing LiDAR renderings as the control signal. Very recently, InverseDVS [65] has proposed a training-free approach that reformulates in-painting as structured latent manipulation in the noise initialization phase of a video diffusion model.

While the shift towards explicit scene reconstruction and test-time optimization has led to high-quality dynamic view synthesis in the *narrow* setting, where camera motion is limited to neighbouring and highly overlapping regions, we experimentally demonstrate that these methods do not generalize to the more challenging *extreme* setting. In contrast, data-driven, implicit approaches are in principle capable of dynamic view synthesis from any view point, but are in

practice limited by the availability of diverse training data. In this work, we address this limitation by (1) combining a wide body of publicly available datasets to train AnyView — the first model capable of synthesizing arbitrary novel views in dynamic, real-world scenes; and (2) proposing a new benchmark, AnyViewBench, to properly evaluate dynamic view synthesis performance in this new setting.

3. Methodology

3.1. Problem Statement

The goal of dynamic view synthesis (DVS) is to create an output video \mathbf{V}_y of an underlying scene as depicted from a chosen virtual viewpoint c_y , given an input video \mathbf{V}_x recorded by a camera with known poses c_x and intrinsics i_x over time. Specifically, we define the input (observed) RGB video as $\mathbf{V}_x \in \mathbb{R}^{T \times H \times W \times 3}$, the target (unobserved) RGB video as $\mathbf{V}_y \in \mathbb{R}^{T \times H \times W \times 3}$, the input camera trajectory as $c_x \in \mathbb{R}^{T \times 4 \times 4}$ with intrinsics $i_x \in \mathbb{R}^{T \times 3 \times 3}$, and the target camera trajectory as $c_y \in \mathbb{R}^{T \times 4 \times 4}$ with intrinsics $i_y \in \mathbb{R}^{T \times 3 \times 3}$. Using a generative model f , we estimate \mathbf{V}_y corresponding to the desired novel viewpoint c_y by drawing from a conditional probability distribution:

$$\mathbf{V}_y \sim P_f(\mathbf{V}_y | \mathbf{V}_x, c_x, i_x, c_y, i_y) \quad (1)$$

The camera parameters c_x, i_x, c_y, i_y represent two sequences of fully specified 6-DoF $SE(3)$ camera poses, ensuring that the task setting is both general and unambiguous. Moreover, there should be some spatial overlap in content between the two perspectives c_x and c_y (even if this overlap is temporally asynchronous), otherwise the conditioning signal loses its relevance.

3.2. Architecture

The task described above involves (1) *synthesis of high-dimensional data* in the form of multiple images, and (2) *considerable uncertainty handling* mainly due to occlusion and ambiguous object motion. These requirements are challenging, but naturally lend themselves to being implemented using the generative video paradigm. Hence, we adopted Cosmos [37], a latent diffusion transformer, as our

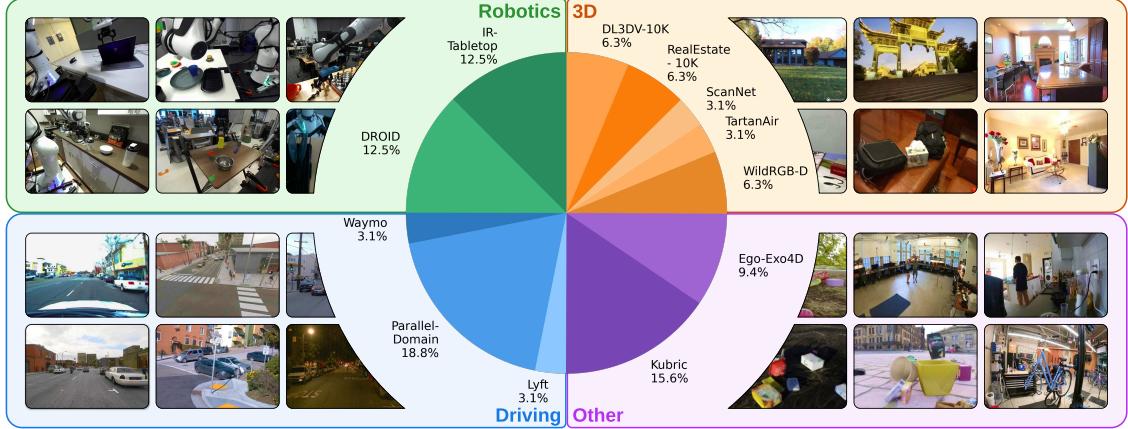


Figure 3. **Overview of our training data mixture.** We train and evaluate AnyView on both single-view and multi-view videos from four domains: *3D*, *Driving*, *Robotics*, and *Other* (see Section 3.3). During training, we perform weighted sampling to ensure each domain is seen equally often, *i.e.* comprises 25% of the batch.

underlying base representation, due to its efficiency, high-quality pretrained checkpoints, and flexible conditioning mechanisms (*e.g.* text, image, and video).

Our proposed AnyView architecture, illustrated in Figure 2, prioritizes simplicity and scalability. Contrary to most state-of-the-art methods [9, 43, 60, 68], we elect to not use warped depth maps as explicit conditioning due to the risk of compounding errors due to depth estimation, and instead rely solely on a learned implicit representation as our rendering mechanism. The reasoning behind this decision is so that we can achieve *unbounded* dynamic view synthesis, that does not require substantial overlap between target and generated videos, thus allowing for more extreme camera motion. We explore this property in our proposed AnyViewBench, outperforming baselines that rely on explicit reprojection mechanisms.

In order to make AnyView 4D-aware and controllable, we feed information about both viewpoints into the network in a structured yet straightforward way. To account for the possible lack of an absolute frame of reference, all camera poses are expected to exist relative to the target viewpoint $c_{y,0}$ at time $t = 0$. In other words, c_y always starts at the “origin”, with $c_{y,0} = I_{4 \times 4}$ mapping to the identity matrix. If this is not the case, a simple change of coordinate system can be done by applying $\tilde{c} = c \cdot c_{y,0}^{-1}$, assuming the camera-to-world extrinsics convention.

First, the given video V_x is compressed into a latent space by a video tokenizer to become $v_x \in \mathbb{R}^{t \times h \times w \times d}$, with spatiotemporal downsampling ratios $T/t = 4$ and $H/h = W/w = 8$, and embedding size $d = 16$. We then encode all camera parameters c_x, i_x, c_y, i_y into a unified *Plücker representation* $P = (r, m)$ [22], which combines extrinsics and intrinsics into a dense map containing per-pixel *ray* vectors r and *moment* vectors $m = r \times o$. This results in two quantities $P_x, P_y \in \mathbb{R}^{T \times H \times W \times 6}$, which are tensors with the same dimensionality as a 6-

channel video, or two 3-channel videos. We can therefore separately tokenize the rays $r \in \mathbb{R}^{T \times H \times W \times 3}$ and moments $m \in \mathbb{R}^{T \times H \times W \times 3}$ the same way as before into $p_x, p_y \in \mathbb{R}^{t \times h \times w \times 2d}$. An interesting property of using Plücker maps instead of direct camera conditioning [50] is the natural handling of non-pinhole camera models, since the dense 3D ray vectors directly capture camera intrinsics in a general, non-parametric way.

Because latent RGB and Plücker tokens from each viewpoint contain information pertaining to the same spatiotemporal region, we merge them via concatenation along the channel dimension, while keeping tokens from separate viewpoints separate. Since there are two viewpoints in total, this results in a sequence of $2 \cdot t \cdot h \cdot w$ tokens, each of length $3 \cdot d$. All tokens are tagged with rotary positional embeddings [47], as well as a unique per-view embedding. After completing all self-attention and cross-attention blocks, the output sequence is the latent RGB video $v_y \in \mathbb{R}^{t \times h \times w \times d}$. During training, these latent tokens are supervised with an \mathcal{L}_2 loss, and during inference they are iteratively denoised, before being decoded into a generated video $V_y \in \mathbb{R}^{T \times H \times W \times 3}$.

3.3. Datasets

Because AnyView does not rely on any explicit conditioning mechanism (*e.g.* intermediate depth maps) to facilitate the rendering of novel viewpoints, it must learn implicit multi-view geometry as well as a wide range of appearance priors, to be able to inpaint and outpaint potentially large unobserved portions of the scene. In order to train such a generalist spatiotemporal representation capable of handling multiple domains, we combined 12 different 4D datasets into our unified training pipeline. Among them is *Kubric-5D*, our newly introduced variation of Kubric-4D [16, 50] that vastly increases the diversity of camera trajectories. We classify our training datasets into four dis-

Benchmark	S/R	Domain	Type	# Cameras	# Episodes	Resolution	Input Cam.	Align Start	Gen. Type
Narrow dynamic view synthesis									
DyCheck iPhone [14]	Real	Hand-Object	4D	2 – 3	5 / 7	Variable @ 288 × 384	Moving	No	0-shot overall
Kubric-4D (gradual) [16]	Sim	Multi-Object	4D	16 (exo only)	20 / 100	13 frames @ 384 × 256	Fixed	Yes	In-dist.
ParDom-4D (gradual) [1]	Sim	Driving	4D	19 (16 exo + 3 ego)	20 / 61	13 frames @ 384 × 256	Variable	Yes	In-dist.
AnyViewBench: In-distribution extreme dynamic view synthesis									
DROID (ID) [29]	Real	Robotics	4D	2 (exo only)	64 / 3,301	29 frames @ 384 × 208	Fixed	No	In-dist.
Ego-Exo4D (ID) [15]	Real	Human Activity	4D	4 – 5 (exo only)	64 / 276	41 frames @ 384 × 208	Fixed	No	In-dist.
LBM	Sim + Real	Robotics	4D	2 (exo only)	64 / 5,988	41 frames @ 336 × 256	Fixed	No	In-dist.
Kubric-4D (direct) [16]	Sim	Multi-Object	4D	16 (exo only)	20 / 100	13 frames @ 384 × 256	Fixed	No	In-dist.
Kubric-5D [16]	Sim	Multi-Object	4D	16 (exo only)	64 / 200	41 frames @ 384 × 256	Variable	No	In-dist.
Lyft [23]	Real	Driving	4D	6 (ego only)	64 / 436	41 frames @ 384 × 320	Variable	No	In-dist.
ParDom-4D (direct) [1]	Sim	Driving	4D	19 (16 exo + 3 ego)	20 / 61	13 frames @ 384 × 256	Variable	No	In-dist.
Waymo [48]	Real	Driving	4D	5 (ego only)	64 / 202	41 frames @ 384 × {176, 256}	Variable	No	In-dist.
AnyViewBench: Zero-shot extreme dynamic view synthesis									
Argoverse [56]	Real	Driving	4D	7 (ego only)	64 / 1,042	41 frames @ {288, 384} × {288, 384}	Variable	No	0-shot dataset
AssemblyHands [38]	Real	Hand-Object	4D	8 (exo only)	20 / 20	41 frames @ 384 × 208	Fixed	No	0-shot domain
DDAD [17]	Real	Driving	4D	6 (ego only)	64 / 200	41 frames @ 384 × 240	Variable	No	0-shot dataset
DROID (OOD) [29]	Real	Robotics	4D	2 (exo only)	64 / 252	29 frames @ 384 × 208	Fixed	No	0-shot station
Ego-Exo4D (OOD) [15]	Real	Human Activity	4D	4 – 5 (exo only)	64 / 408	41 frames @ 384 × 208	Fixed	No	0-shot activity/site

Table 1. Testing datasets. We evaluate on several benchmarks that cover both *narrow* and *extreme* settings. We define **AnyViewBench** as a multi-faceted benchmark focusing on the latter category, setting a new standard for consistent dynamic view synthesis in challenging settings. Test splits are capped at 64 per dataset by means of uniform subsampling. *Exo(centric)* refers to inward-facing viewpoints from cameras outside the scene, whereas *ego(centric)* refers to outward-facing viewpoints close to the subject of interest (e.g. a vehicle). *Input Cam.* refers to what the camera characteristic of the observed video (i.e. static vs dynamic). *Align Start* specifies whether the output trajectory starts at the same initial frame as the input. The rightmost column (*Generalization Type*) qualitatively denotes how large the distribution shift is relative to the AnyView training mixture.

tinct quadrants: *Robotics*, *Driving*, *3D*, and *Other*. A visual overview is illustrated in Figure 3, and more details are provided in the supplementary material. To the best of our knowledge, this data mixture covers a significant portion of publicly available multi-view video datasets. We leave the inclusion of additional 4D datasets [41, 42, 71] to future work.

3.4. Implementation Details

We train AnyView for 40,000 iterations on 64 NVIDIA H200 GPUs at a global batch size of 512. We apply curriculum learning with increasing resolution: first we train at a largest image dimension of 384 for 30,000 steps, before finetuning at a largest image dimension of 576. The initial learning rate is $5 \cdot 10^{-5}$, and drops smoothly to $1 \cdot 10^{-5}$ according to a cosine schedule. All experiments are performed with the *Cosmos-Predict2-2B-Video2World* [36] model, starting from their pretrained network, which has around 2 billion parameters. We disable language conditioning, since it is not relevant to our task setting. Furthermore, in order to properly combine datasets with varying physical scales, we divide the translation vectors of all cameras $\{c_x, c_y\}$ by a carefully chosen per-dataset normalization constant to ensure the resulting Plücker values always fall in the range $[-1, 1]$, clipping pixels as needed.

4. Experiments

4.1. Evaluation Challenges

As the field is evolving, many existing DVS benchmarks are beginning to **lack difficulty**, containing scenes with minimal object motion and modest camera transformations [14, 33, 50, 67]. Qualitative results are often demonstrated on camera trajectories with variations of only about 10–30 degrees relative to the center of the scene [4, 50, 53, 66, 69]. Consequently, the heavy lifting of inpainting large occlusions is mostly avoided, making it unclear to which extent these models learn robust, multi-view consistent 4D representations. These efforts are further complicated by a **lack of standardization**, which can be partially attributed to the inherent complexity of DVS: describing the task is insufficient to define a path towards practical execution. These design choices include but are not limited to: video resolution, number of frames, camera controllability, the space of possible camera transformations, and so on.

4.2. Benchmarks

We first consider three popular DVS benchmarks that can be classified as falling into the “narrow” regime. Then, to address the aforementioned concerns, we propose *AnyViewBench*, which substantially pushes models into the more challenging “extreme” regime.

DyCheck iPhone (narrow DVS). The iPhone dataset [14] is a small collection of high-quality, real-world, multi-view videos of easy-to-moderate difficulty established to measure DVS fidelity. Following previous works [9], that have



Figure 4. **AnyView in-domain DVS results on Kubric-4D** (left) and **Pardom-4D** (right). We show the first and last frame of each video. The scene layout is generally preserved very well, despite drastic viewpoint changes and/or heavy occlusion from the input vantage point.

Method	PSNR↑	SSIM↑	LPIPS↓	TTO	Aux.	0-shot	Time
DyCheck iPhone [14]							
ShapeOfMotion [†] [53]	16.72	0.630	0.450	✓	D P T	—	~1h
CogNVS [†] [9]	16.94	0.449	0.598	✓	D(GT)	—	~1h
GEN3C [*] [43]	10.13	0.175	0.695	—	D	✓	~15m
TrajAttn [*] [60]	10.30	0.181	0.682	—	D	✓	~10m
TrajCrafter [†] [68]	14.24	0.417	0.519	—	D	✓	~10s
GCD [*] [50]	11.43	0.247	0.728	—	—	✓	~10s
AnyView	13.47	0.295	0.550	—	—	✓	~10s
Kubric-4D (gradual) [16]							
CogNVS [†] [9]	22.63	0.760	0.232	✓	D(GT)	—	~1h
ReCapture [†] [69]	20.92	0.596	0.402	✓	D	—	~15m
GEN3C [†] [43]	19.41	0.630	0.290	—	D	✓	~15m
TrajAttn [*] [60]	15.73	0.404	0.530	—	D	✓	~5m
TrajCrafter [‡] [9]	20.93	0.730	0.257	—	D	✓	~10s
GCD [*] [50]	20.42	0.581	0.405	—	—	—	~10s
AnyView	21.21	0.644	0.358	—	—	—	~10s
ParDom-4D (gradual) [1]							
CogNVS [†] [9]	24.34	0.797	0.302	✓	D(GT)	—	~1h
GEN3C [*] [43]	18.40	0.528	0.542	—	D	✓	~15m
TrajAttn [*] [60]	20.03	0.566	0.518	—	D	✓	~5m
TrajCrafter [‡] [9]	21.46	0.719	0.342	—	D	✓	~10s
GCD [*] [50]	24.75	0.724	0.355	—	—	—	~10s
AnyView	26.29	0.758	0.320	—	—	—	~10s

Table 2. **Narrow DVS results.** We compare against several state-of-the-art baselines, including those using test-time optimization (TTO) and auxiliary networks (Aux.) for depth (D), poses (P), and/or 2D point tracks (T). The inference runtime assumes that a video was not observed before, and thus includes a test-time optimization stage if present. Results reported by: [†]original paper; [‡]another paper (cited); ^{*}computed by us.

pointed out that the provided camera poses are not very accurate, we compute corrected extrinsics using MoSca [31].

Kubric-4D and ParDom-4D (narrow + extreme DVS). The GCD [50] paper introduced two synthetic datasets for DVS training and evaluation, based on the Kubric [16] and ParallelDomain [1] simulation environments.

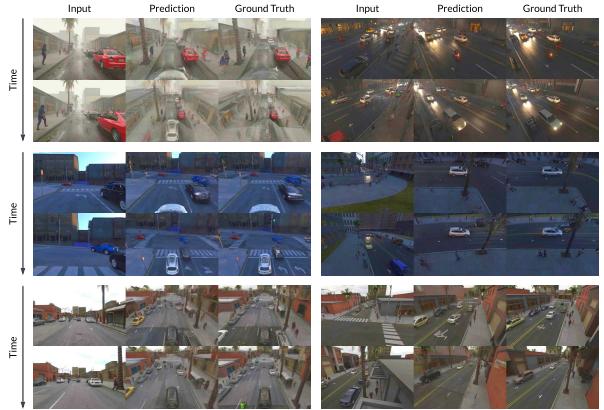


Figure 5. **Results on DyCheck iPhone (0-shot narrow DVS).** While these scenes are not highly dynamic, they do contain subtle, intricate motions and hand-object interactions.

AnyViewBench (extreme DVS). We introduce AnyViewBench, a multi-faceted benchmark that covers datasets across multiple domains (driving, robotics, and human activities), as shown in Table 1. The camera motion patterns range from simple (fixed or linear) to complex (e.g. highly non-linear trajectories, changing intrinsics, etc.). To promote rigorous evaluation, we provide synchronized videos from at least two separate viewpoints for each episode, with well-defined details such as spatial resolution and number of frames, such that ground truth metrics can be calculated in a straightforward manner. For all *in-distribution* datasets, we separate roughly 10% to serve as validation, and for both *in-distribution* and *zero-shot* datasets we curate smaller subsets to serve as official test splits. Moreover, two DROID stations (GuptaLab, ILIAD), as well as certain EgoExo4D institutions (FAIR, NUS) and activities (CPR, Guitar), are held out to serve as *zero-shot* evaluation. More information about AnyViewBench can be found in the supplementary material, and we will release it upon publication.

4.3. Baselines

Most current DVS methods face key limitations: the input video must be captured from a strictly *static* camera [50], or from a strictly *dynamic* camera [9, 43], or both input and output videos must start from the same position [4, 60, 68], or the camera controlling mechanism has limited degrees

Dataset	GCD [50]			TrajAttn [60]			GEN3C [43]			TrajCrafter [68]			CogNVS [9]			Ours (AnyView)		
	PSNR↑ SSIM↑ LPIPS↓			PSNR↑ SSIM↑ LPIPS↓			PSNR↑ SSIM↑ LPIPS↓			PSNR↑ SSIM↑ LPIPS↓			PSNR↑ SSIM↑ LPIPS↓			PSNR↑ SSIM↑ LPIPS↓		
In-distribution																		
DROID (ID) [29]	10.18	0.255	0.688	9.93	0.247	0.671	9.62	0.211	0.666	10.45	0.257	0.620	9.44	0.281	0.634	14.47	0.445	0.472
EgoExo4D (ID) [15]	12.10	0.255	0.670	11.64	0.234	0.646	11.69	0.222	0.643	11.33	0.195	0.642	10.80	0.241	0.670	18.14	0.531	0.379
LBM	12.59	0.398	0.694	13.47	0.421	0.614	13.48	0.449	0.581	13.68	0.447	0.537	13.30	0.453	0.548	17.94	0.649	0.348
Kubric-4D (direct) [16]	17.57	0.477	0.512	13.47	0.320	0.607	13.59	0.341	0.599	14.14	0.294	0.592	12.55	0.329	0.601	18.38	0.441	0.362
Kubric-5D	13.83	0.391	0.644	13.25	0.360	0.628	13.10	0.327	0.627	13.30	0.287	0.625	12.18	0.318	0.643	17.18	0.468	0.428
Lyft [23]	8.72	0.335	0.697	8.33	0.273	0.621	8.43	0.286	0.634	8.73	0.251	0.621	8.34	0.319	0.628	15.37	0.564	0.371
ParDom-4D (direct) [1]	22.67	0.656	0.457	16.91	0.445	0.610	16.64	0.478	0.590	18.23	0.475	0.586	18.36	0.499	0.564	24.26	0.688	0.351
Waymo [48]	12.93	0.393	0.647	12.55	0.350	0.613	12.98	0.350	0.600	12.66	0.312	0.593	13.27	0.377	0.594	16.52	0.477	0.480
Average	13.95	0.400	0.623	12.44	0.331	0.626	12.44	0.333	0.617	12.82	0.315	0.602	12.28	0.352	0.610	17.78	0.533	0.399
Zero-shot																		
Argoverse [56]	11.45	0.403	0.682	10.62	0.317	0.665	10.52	0.319	0.680	10.67	0.325	0.621	10.76	0.360	0.610	12.38	0.399	0.587
AssemblyHands [38]	9.77	0.262	0.759	9.97	0.266	0.736	9.86	0.237	0.732	11.45	0.248	0.701	9.93	0.281	0.701	11.21	0.291	0.688
DDAD [17]	9.81	0.278	0.660	9.16	0.244	0.620	9.35	0.259	0.600	10.73	0.300	0.558	10.81	0.355	0.572	11.44	0.341	0.519
DROID (OOD) [29]	11.81	0.315	0.690	10.83	0.320	0.678	10.56	0.276	0.674	11.37	0.339	0.614	10.48	0.358	0.632	12.34	0.422	0.601
EgoExo4D (OOD) [15]	11.98	0.239	0.668	11.31	0.203	0.653	11.40	0.193	0.651	11.27	0.180	0.647	10.52	0.227	0.683	13.30	0.297	0.562
Average	10.96	0.299	0.692	10.38	0.270	0.670	10.34	0.257	0.667	11.10	0.279	0.628	10.50	0.316	0.640	12.03	0.350	0.591

Table 3. **Extreme DVS results (AnyViewBench).** Note that *in-distribution* datasets are part of AnyView’s training mixture, but might be zero-shot for some of the baselines, hence we provide these results for completeness. For qualitative comparison, please refer to Figure 1.



Figure 6. **AnyView extreme DVS results on driving** (left) and **robotics** (right) benchmarks. We show both in-domain and zero-shot results. For driving videos, we focus on the three frontal cameras, whereas for robotics, we focus on all scene cameras.

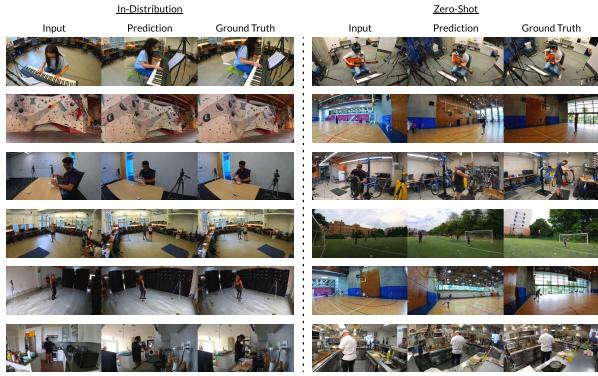
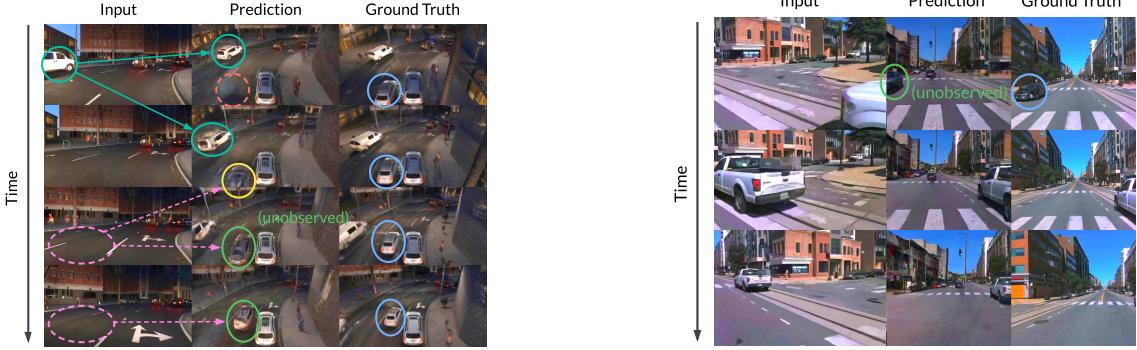


Figure 7. **AnyView extreme DVS results on Ego-Exo4D.** We show both in-domain and zero-shot results. Note that in the zero-shot case, the background often has to be “guessed” from the other camera viewpoint, but the inpainted regions (see *e.g.* basketball, soccer) integrate harmoniously with the rest of the scene.

of freedom [4, 50, 68]. As a result, methods that excel in certain conditions might be incompatible with slightly

different evaluation settings, hindering standardized evaluation across multiple benchmarks. More information detailing all prior works we considered as baselines can be found in the supplementary material. Most of these models already evaluate on at least a subset of the “narrow” benchmarks, but we additionally evaluate them (doing our best effort to project down to and accommodate the space of supported camera transformations as needed) on AnyViewBench, which embodies the “extreme” benchmarks. Re-CamMaster [4] was not evaluated because it does not support arbitrary camera trajectories, and InverseDVS [66] was not evaluated because there was no working released code at the time of submission. When evaluating baseline methods that require depth estimation to render reprojected images, we use DepthAnythingV2 [63] and tune the maximum depth parameter for each dataset to achieve the best alignment between reprojected and ground truth images.



(a) **Headlight reflections.** While the **white van** is partially visible in the first few frames, and thus gets depicted accurately in the remainder of the video, the **white car** is never observed in *any* input frame. Instead, AnyView appears to pick up on the **headlights reflecting on the road**. Although the **reconstructed car** does not have the correct appearance, the model indirectly estimates its trajectory by tracking the reflection over time.

(b) **Driving behavior.** This zero-shot ArgoVerse scene depicts the ego vehicle pausing for a moment, and then turning left. The model correctly hallucinates the **black car** passing by on the left of the generated video, despite never observing it, presumably based on the suspicion that the driver must be waiting at the green light because of oncoming traffic before executing the unprotected left turn.

Figure 8. **Examples of advanced reasoning within AnyView**, as a way to indirectly guide generation in unobserved parts of the scene.

4.4. Results

Following standard convention, we report DVS results in terms of PSNR (dB), SSIM, and LPIPS (VGG), averaged over all frames in the generated video. Note that these metrics can only attest to how similar generated predictions are to the ground truth, but not necessarily how realistic and plausible they are when the true underlying scene cannot be fully known due to lack of overlap between viewpoints.

Quantitative results on existing narrow DVS benchmarks are reported in Table 2, with qualitative results in Figures 4 and 5. For completeness, we also include metrics as reported by other papers, as well as evaluate the baselines ourselves when possible. AnyView outperforms GCD [50], the only baseline that does not require explicit depth estimation or reprojection, by a large margin, and compares favorably with explicit depth reprojection methods — and those that require expensive test-time-optimization — in most metrics. This *narrow* setting (i.e., large overlapping regions with small viewpoint changes) is particularly well-suited for such methods, since a lot of information can be directly transferred across viewpoints, and the model is tasked solely with inpainting the missing regions.

Next, we report results in *extreme* DVS setting using AnyViewBench, with quantitative results in Table 3 and illustrations in Figures 6 and 7. These scenarios are much more challenging, since they require implicit 4D understanding to ensure spatiotemporal consistency. For example, in real-world driving, the amount of spatial overlap between neighboring cameras is generally small, meaning that when the model is prompted with generating the front-left view based solely on the front view (or vice-versa), it has to plausibly infer the majority of the scene based on little information. However, if the ego vehicle is moving, information is able to eventually “leak” into other views and can

be propagated across the entire sequence, further limiting the space of “correct” generations.

In the upper left scenario in Figure 6, the red car arriving at the intersection is predicted on the left view *before* it is visible in the input front view, showing that AnyView has learned to maintain spatiotemporal consistency, leading to improved performance in areas that otherwise would be ill-defined. A related behavior is also observed in the left examples of Figure 7, where AnyView leverages its foundational knowledge to infer how a basketball court or soccer field should look like from different perspectives. Moreover, in Figure 8 we show anecdotal examples of AnyView leveraging subtle visual cues to improve generation accuracy in unobserved areas, as evidence of advanced common sense and spatiotemporal reasoning.

Implicitly learning these useful spatiotemporal properties in a data-driven way enables AnyView to produce more realistic and physically plausible representations of real-world scenarios compared to all baselines. As shown in Figure 1, while methods that rely on potentially inaccurate depth reprojection (e.g. TrajAttn and GEN3C) struggle when starting from target poses away from input poses, AnyView successfully generates smooth, consistent target scenes regardless of camera positioning. Similarly, AnyView is able to accurately outpaint much larger unobserved portions of the scene compared to methods trained mostly for limited inpainting (e.g. TrajCrafter and CogNVS). As a consequence of these useful properties, we achieve state of the art zero-shot DVS performance on AnyViewBench, outperforming all other baseline methods by a significant margin across all considered datasets.

5. Discussion

In this paper, we propose *AnyView*, a generalist dynamic view synthesis framework targeting extreme camera dis-

placements. We also contribute *AnyViewBench*, a well-rounded benchmark that focuses on highly challenging scenarios from various domains, showing that AnyView significantly outperforms baselines in such settings with large camera displacement and limited overlap between views. We hope that this work provides a useful building block towards improving video foundation models and 4D representations, with potential applications in dynamic scene reconstruction, world models, robotics, self-driving, and more.

References

- [1] Parallel domain. <https://paralleldomain.com/>, 2024. 5, 6, 7, 3
- [2] Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025. 2
- [3] Sherwin Bahmani, Ivan Skorokhodov, Guocheng Qian, Ali-aksandr Siarohin, Willi Menapace, Andrea Tagliasacchi, David B Lindell, and Sergey Tulyakov. Ac3d: Analyzing and improving 3d camera control in video diffusion transformers. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22875–22889, 2025. 3
- [4] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, et al. Recammaster: Camera-controlled generative rendering from a single video. *arXiv preprint arXiv:2503.11647*, 2025. 3, 5, 6, 7, 4
- [5] Renee Baillargeon. Object permanence in 31/2-and 41/2-month-old infants. *Developmental psychology*, 23(5):655, 1987. 2
- [6] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets, 2023. 4
- [7] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 2
- [8] Neil Burgess. Spatial memory: how egocentric and allocentric combine. *Trends in cognitive sciences*, 10(12):551–557, 2006. 2
- [9] Kaihua Chen, Tarasha Khurana, and Deva Ramanan. Reconstruct, inpaint, finetune: Dynamic novel-view synthesis from monocular videos. 2025. 2, 3, 4, 5, 6, 7
- [10] Lawrence Yunliang Chen, Chenfeng Xu, Karthik Dharmajan, Zubair Irshad, Richard Cheng, Kurt Keutzer, Masayoshi Tomizuka, Quan Vuong, and Ken Goldberg. Rovi-aug: Robot and viewpoint augmentation for cross-embodiment robot learning. In *Conference on Robot Learning (CoRL)*, Munich, Germany, 2024. 2
- [11] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 1, 3
- [12] James J DiCarlo and David D Cox. Untangling invariant object recognition. *Trends in cognitive sciences*, 11(8):333–341, 2007. 2
- [13] Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. Dynamic view synthesis from dynamic monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5712–5721, 2021. 2
- [14] Hang Gao, Ruilong Li, Shubham Tulsiani, Bryan Russell, and Angjoo Kanazawa. Dynamic novel-view synthesis: A reality check. In *NeurIPS*, 2022. 2, 5, 6
- [15] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, Eugene Byrne, Zach Chavis, Joya Chen, Feng Cheng, Fu-Jen Chu, Sean Crane, Avijit Dasgupta, Jing Dong, Maria Escobar, Cristhian Forigua, Abrham Gebreselasie, Sanjay Haresh, Jing Huang, Md Mohaiminul Islam, Suyog Jain, Rawal Khirodkar, Devansh Kukreja, Kevin J Liang, Jia-Wei Liu, Sagnik Majumder, Yongsen Mao, Miguel Martin, Effrosyni Mavroudi, Tushar Nagarajan, Francesco Ragusa, Santhosh Kumar Ramakrishnan, Luigi Seminara, Arjun Somayazulu, Yale Song, Shan Su, Zihui Xue, Edward Zhang, Jin Xu Zhang, Angela Castillo, Changan Chen, Xinzhu Fu, Ryosuke Furuta, Cristina Gonzalez, Prince Gupta, Jiabo Hu, Yifei Huang, Yiming Huang, Leslie Khoo, Anush Kumar, Robert Kuo, Sach Lakhavani, Miao Liu, Mi Luo, Zhengyi Luo, Brighid Meredith, Austin Miller, Oluwatuminu Oguntola, Xiaqing Pan, Penny Peng, Shraman Pramanick, Merey Ramazanova, Fiona Ryan, Wei Shan, Kiran Somasundaram, Chenan Song, Audrey Southerland, Masatoshi Tateno, Huiyu Wang, Yuchen Wang, Takuma Yagi, Mingfei Yan, Xitong Yang, Zecheng Yu, Shengxin Cindy Zha, Chen Zhao, Ziwei Zhao, Zhifan Zhu, Jeff Zhuo, Pablo Arbelaez, Gedas Bertasius, David Crandall, Dima Damen, Jakob Engel, Giovanni Maria Farinella, Antonino Furnari, Bernard Ghanem, Judy Hoffman, C. V. Jawahar, Richard Newcombe, Hyun Soo Park, James M. Rehg, Yoichi Sato, Manolis Savva, Jianbo Shi, Mike Zheng Shou, and Michael Wray. Ego-exo4d: Understanding skilled human activity from first- and third-person perspectives, 2024. 5, 7, 1, 3, 4
- [16] 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. 2022. 4, 5, 6, 7, 1, 3
- [17] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *CVPR*, 2020. 5, 7, 3, 4
- [18] Vitor Guizilini, Jie Li, Rares Ambrus, and Adrien Gaidon. Geometric unsupervised domain adaptation for semantic segmentation. In *ICCV*, 2021. 1
- [19] Vitor Guizilini, Kuan-Hui Lee, Rares Ambrus, and Adrien Gaidon. Learning optical flow, depth, and scene flow without real-world labels. *IEEE Robotics and Automation Letters*, 2022. 1

[20] Yanjiang Guo, Lucy Xiaoyang Shi, Jianyu Chen, and Chelsea Finn. Ctrl-world: A controllable generative world model for robot manipulation, 2025. 2

[21] 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 preprint arXiv:2404.02101*, 2024. 2

[22] W. V. D. Hodge and Daniel Pedoe. *Methods of Algebraic Geometry, Volume 1*. Cambridge University Press, London/New York, 1947. Original Publication. 4

[23] John Houston, Guido Zuidhof, Luca Bergamini, Yawei Ye, Long Chen, Ashesh Jain, Sammy Omari, Vladimir Iglovikov, and Peter Ondruska. One thousand and one hours: Self-driving motion prediction dataset. In *CoRL*, pages 409–418, 2020. 5, 7, 1, 3, 4

[24] Wenbo Hu, Xiangjun Gao, Xiaoyu Li, Sijie Zhao, Xiaodong Cun, Yong Zhang, Long Quan, and Ying Shan. Depthcrafter: Generating consistent long depth sequences for open-world videos. In *CVPR*, 2025. 5

[25] Jiaxin Huang, Sheng Miao, Bangbang Yang, Yuwen Ma, and Yiyi Liao. Vivid4d: Improving 4d reconstruction from monocular video by video inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12592–12604, 2025. 2

[26] Jiahui Huang, Qunjie Zhou, Hesam Rabeti, Aleksandr Krovko, Huan Ling, Xuanchi Ren, Tianshang Shen, Jun Gao, Dmitry Slepichev, Chen-Hsuan Lin, Jiawei Ren, Kevin Xie, Joydeep Biswas, Laura Leal-Taixe, and Sanja Fidler. Vipe: Video pose engine for 3d geometric perception. In *NVIDIA Research Whitepapers*, 2025. 5

[27] Muhammad Zubair Irshad, Vitor Guizilini, Alexander Khazatsky, and Karl Pertsch. Scaling-up automatic camera calibration for droid dataset: A study using foundation models and existing deep-learning tools. medium.com/p/4ddfc45361d3, 2024. Medium blog post. 1

[28] Daniel Kahneman, Anne Treisman, and Brian J Gibbs. The reviewing of object files: Object-specific integration of information. *Cognitive psychology*, 24(2):175–219, 1992. 2

[29] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, Peter David Fagan, Joey Hejna, Masha Itkina, Marion Lepert, Yecheng Jason Ma, Patrick Tree Miller, Jimmy Wu, Suneel Belkhale, Shivin Dass, Huy Ha, Arhan Jain, Abraham Lee, Youngwoon Lee, Marius Memmel, Sungjae Park, Ilija Radosavovic, Kaiyuan Wang, Albert Zhan, Kevin Black, Cheng Chi, Kyle Beltran Hatch, Shan Lin, Jingpei Lu, Jean Mercat, Abdul Rehman, Pannag R Sanketi, Archit Sharma, Cody Simpson, Quan Vuong, Homer Rich Walke, Blake Wulfe, Ted Xiao, Jonathan Heewon Yang, Arefeh Yavary, Tony Z. Zhao, Christopher Agia, Rohan Baijal, Mateo Guaman Castro, Daphne Chen, Qiuyu Chen, Trinity Chung, Jaimyn Drake, Ethan Paul Foster, Jensen Gao, Vitor Guizilini, David Antonio Herrera, Minho Heo, Kyle Hsu, Jiaheng Hu, Muhammad Zubair Irshad, Donovan Jackson, Charlotte Le, Yunshuang Li, Kevin Lin, Roy Lin, Zehan Ma, Abhiram Maddukuri, Suvir Mirchandani, Daniel Morton, Tony Nguyen, Abigail O'Neill, Rosario Scalise, Derick Seale, Victor Son, Stephen Tian, Emi Tran, Andrew E. Wang, Yilin Wu, Annie Xie, Jingyun Yang, Patrick Yin, Yunchu Zhang, Osbert Bastani, Glen Berseth, Jeannette Bohg, Ken Goldberg, Abhinav Gupta, Abhishek Gupta, Dinesh Jayaraman, Joseph J Lim, Jitendra Malik, Roberto Martín-Martín, Subramanian Ramamoorthy, Dorsa Sadigh, Shuran Song, Jiajun Wu, Michael C. Yip, Yuke Zhu, Thomas Kollar, Sergey Levine, and Chelsea Finn. Droid: A large-scale in-the-wild robot manipulation dataset. 2024. 5, 7, 1, 3, 4

[30] Hidehiko Komatsu. The neural mechanisms of perceptual filling-in. *Nature reviews neuroscience*, 7(3):220–231, 2006. 2

[31] Jiahui Lei, Yijia Weng, Adam W Harley, Leonidas Guibas, and Kostas Daniilidis. Mosca: Dynamic gaussian fusion from casual videos via 4d motion scaffolds. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6165–6177, 2025. 6

[32] Junbang Liang, Ruoshi Liu, Ege Ozguroglu, Sruthi Sudhakar, Achal Dave, Pavel Tokmakov, Shuran Song, and Carl Vondrick. Dreamitate: Real-world visuomotor policy learning via video generation. *arXiv preprint arXiv:2406.16862*, 2024. 2

[33] Kai-En Lin, Lei Xiao, Feng Liu, Guowei Yang, and Ravi Ramamoorthi. Deep 3d mask volume for view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1749–1758, 2021. 5

[34] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. DI3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22160–22169, 2024. 1, 3

[35] Bence Nanay. The importance of amodal completion in everyday perception. *i-Perception*, 9(4):2041669518788887, 2018. 2

[36] NVIDIA. Cosmos-predict2: Diffusion-based world foundation models for physics-aware image and video generation. <https://github.com/nvidia-cosmos/cosmos-predict2>, 2025. Accessed: 2025-11-06. 5

[37] NVIDIA, Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, Prithvijit Chattopadhyay, Mike Chen, Yongxin Chen, Yu Chen, Shuai Cheng, Yin Cui, Jenna Diamond, Yifan Ding, Jiaoqiao Fan, Linxi Fan, Liang Feng, Francesco Ferroni, Sanja Fidler, Xiao Fu, Ruiyuan Gao, Yunhao Ge, Jinwei Gu, Aryaman Gupta, Siddharth Gururani, Imad El Hanafi, Ali Hassani, Zekun Hao, Jacob Huffman, Joel Jang, Pooya Jannaty, Jan Kautz, Grace Lam, Xuan Li, Zhaoshuo Li, Maosheng Liao, Chen-Hsuan Lin, Tsung-Yi Lin, Yen-Chen Lin, Huan Ling, Ming-Yu Liu, Xian Liu, Yifan Lu, Alice Luo, Qianli Ma, Hanzi Mao, Kaichun Mo, Seungjun Nah, Yashraj Narang, Abhijeet Panaskar, Lindsey Pavao, Trung Pham, Morteza Ramezanali, Fitsum Reda, Scott Reed, Xuanchi Ren, Haonan Shao, Yue Shen, Stella Shi, Shuran Song, Bartosz Stefaniak, Shangkun Sun, Shitao Tang, Sameena Tasmeen, Lyne Tchapmi, Wei-Cheng Tseng, Jibin Varghese, Andrew Z. Wang, Hao Wang,

Haoxiang Wang, Heng Wang, Ting-Chun Wang, Fangyin Wei, Jiashu Xu, Dinghao Yang, Xiaodong Yang, Haotian Ye, Seonghyeon Ye, Xiaohui Zeng, Jing Zhang, Qinsheng Zhang, Kaiwen Zheng, Andrew Zhu, and Yuke Zhu. World simulation with video foundation models for physical ai, 2025. 2, 3, 4

[38] Takehiko Ohkawa, Kun He, Fadime Sener, Tomas Hodan, Luan Tran, and Cem Keskin. AssemblyHands: towards egocentric activity understanding via 3d hand pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12999–13008, 2023. 5, 7

[39] Jing-Cheng Pang, Nan Tang, Kaiyuan Li, Yuting Tang, Xin-Qiang Cai, Zhen-Yu Zhang, Gang Niu, Masashi Sugiyama, and Yang Yu. Learning view-invariant world models for visual robotic manipulation. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[40] Soojin Park, Helene Intraub, Do-Joon Yi, David Widders, and Marvin M Chun. Beyond the edges of a view: boundary extension in human scene-selective visual cortex. *Neuron*, 54(2):335–342, 2007. 2

[41] Alexander Raistrick, Lahav Lipson, Zeyu Ma, Lingjie Mei, Mingzhe Wang, Yiming Zuo, Karhan Kayan, Hongyu Wen, Beining Han, Yihan Wang, Alejandro Newell, Hei Law, Ankit Goyal, Kaiyu Yang, and Jia Deng. Infinite photorealistic worlds using procedural generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12630–12641, 2023. 5

[42] Alexander Raistrick, Lingjie Mei, Karhan Kayan, David Yan, Yiming Zuo, Beining Han, Hongyu Wen, Meenal Parakh, Stamatis Alexandropoulos, Lahav Lipson, Zeyu Ma, and Jia Deng. Infinigen indoors: Photorealistic indoor scenes using procedural generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21783–21794, 2024. 5

[43] Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 3, 4, 6, 7, 1, 5

[44] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2

[45] F. Sener, D. Chatterjee, D. Sheleporov, K. He, D. Singhania, R. Wang, and A. Yao. Assembly101: A large-scale multi-view video dataset for understanding procedural activities. *CVPR* 2022. 5

[46] Roger N Shepard and Jacqueline Metzler. Mental rotation of three-dimensional objects. *Science*, 171(3972):701–703, 1971. 2

[47] Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2021. 4

[48] 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. 5, 7, 1, 3, 4

[49] Stephen Tian, Blake Wulfe, Kyle Sargent, Katherine Liu, Sergey Zakharov, Vitor Guizilini, and Jiajun Wu. View-invariant policy learning via zero-shot novel view synthesis. *arXiv preprint arXiv:2409.03685*, 2024. 2

[50] Basile Van Hoorick, Rundi Wu, Ege Ozguroglu, Kyle Sargent, Ruoshi Liu, Pavel Tokmakov, Achal Dave, Changxi Zheng, and Carl Vondrick. Generative camera dolly: Extreme monocular dynamic novel view synthesis. *ECCV*, 2024. 3, 4, 5, 6, 7, 8, 1, 2

[51] Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitry Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion. In *European Conference on Computer Vision*, pages 439–457. Springer, 2024. 2

[52] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 2, 4

[53] Qianqian Wang, Vickie Ye, Hang Gao, Weijia Zeng, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4d reconstruction from a single video. In *International Conference on Computer Vision (ICCV)*, 2025. 2, 5, 6

[54] 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. 1, 3

[55] Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, Jia-gang Zhu, and Jiwen Lu. Drivedreamer: Towards real-world-drive world models for autonomous driving. In *European conference on computer vision*, pages 55–72. Springer, 2024. 2

[56] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021)*, 2021. 5, 7, 4

[57] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20310–20320, 2024. 2

[58] Rundi Wu, Ruiqi Gao, Ben Poole, Alex Trevithick, Changxi Zheng, Jonathan T Barron, and Aleksander Holynski. Cat4d: Create anything in 4d with multi-view video diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26057–26068, 2025. 2

[59] Hongchi Xia, Yang Fu, Sifei Liu, and Xiaolong Wang. Rgbd objects in the wild: Scaling real-world 3d object learning from rgb-d videos, 2024. 1, 3

[60] Zeqi Xiao, Wenqi Ouyang, Yifan Zhou, Shuai Yang, Lei Yang, Jianlou Si, and Xingang Pan. Trajectory attention for fine-grained video motion control. In *ICLR*, 2025. 3, 4, 6, 7, 5

[61] Yunzhi Yan, Zhen Xu, Haotong Lin, Haian Jin, Haoyu Guo, Yida Wang, Kun Zhan, Xianpeng Lang, Hujun Bao, Xiaowei Zhou, and Sida Peng. Streetcrafter: Street view synthesis with controllable video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3

[62] Jingyun Yang, Isabella Huang, Brandon Vu, Max Barjacharya, Rika Antonova, and Jeannette Bohg. Mobi- pi: Mobilizing your robot learning policy. *arXiv preprint arXiv:2505.23692*, 2025. 2

[63] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv:2406.09414*, 2024. 7, 5

[64] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 2, 4

[65] Hidir Yesiltepe and Pinar Yanardag. Dynamic view synthesis as an inverse problem. In *NeurIPS*, 2025. 3

[66] Hidir Yesiltepe and Pinar Yanardag. Dynamic view synthesis as an inverse problem, 2025. 2, 5, 7, 4

[67] Jae Shin Yoon, Kihwan Kim, Orazio Gallo, Hyun Soo Park, and Jan Kautz. Novel view synthesis of dynamic scenes with globally coherent depths from a monocular camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5336–5345, 2020. 5

[68] Mark YU, Wenbo Hu, Jinbo Xing, and Ying Shan. Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models. In *ICCV*, 2025. 2, 3, 4, 6, 7, 1, 5

[69] David Junhao Zhang, Roni Paiss, Shiran Zada, Nikhil Kannan, David E Jacobs, Yael Pritch, Inbar Mosseri, Mike Zheng Shou, Neal Wadhwa, and Nataniel Ruiz. Recapture: Generative video camera controls for user-provided videos using masked video fine-tuning. *arXiv preprint arXiv:2411.05003*, 2024. 5, 6

[70] Guangcong Zheng, Teng Li, Rui Jiang, Yehao Lu, Tao Wu, and Xi Li. Cami2v: Camera-controlled image-to-video diffusion model. *arXiv preprint arXiv:2410.15957*, 2024. 2

[71] 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. 5

[72] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. In *SIGGRAPH*, 2018. 1, 3

AnyView: Synthesizing Any Novel View in Dynamic Scenes

Supplementary Material

A. Uncertainty Analysis

Figure 9 showcases how AnyView represents and expresses uncertainty. We calculate this by running the diffusion model multiple times to collect independent samples from the conditional distribution, and plotting the per-pixel diversity between these predictions as a spatial heatmap. Each generation is conditioned on the same input signals, and represents a possible version of what the other viewpoint *could* look like. Even if these outputs are not technically correct, due to the inherent ambiguity of the task at hand, they are still reasonable, realistic, and self-consistent, demonstrating that AnyView learns a powerful probabilistic representation that encodes the natural multimodality of unobserved parts of the world.

B. Additional Qualitative Results

We complement the qualitative results depicted in the main paper with the following:

Figure 10 compares the performance of AnyView against GCD [50] over increasingly wide horizontal camera displacements, showing that AnyView maintains better spatio-temporal consistency over large viewpoint changes.

Figure 11 shows top-down view synthesis on real-world (DDAD) driving scenes, where we also compare against the GCD baseline. This effectively tests each model’s sim-to-real trajectory generalization capability, since the only training videos corresponding to similar viewpoint configurations (albeit still not the same) come from synthetic data (*ParallelDomain*).

Moreover, we include all figures present in the paper as videos in the project webpage: tri-ml.github.io/AnyView. The highly encourage the reader to browse these results, since it is difficult otherwise to communicate 4D results through 2D PDF files.

C. Training Datasets

Here, we provide additional details about the AnyView training mixture, also summarized in Table 4. For all training datasets, we randomly selected around 10% of sequences to serve as *in-distribution* validation, from which many of the official AnyViewBench test splits were curated.

- **Driving:** Most autonomous driving rigs have a set of well-calibrated RGB cameras mounted around the vehicle, providing plenty of real-world, *egocentric* (outward-facing), temporally synchronized video footage. We additionally capitalize on synthetic data to provide *exocentric* (inward-facing) viewpoints that otherwise do

not naturally occur in such datasets. For training, we use the *Woven Planet (Lyft) Level 5* [23], *ParallelDomain* [18, 19, 50], and *Waymo Open (Perception)* [48] datasets.

- **Robotics:** To enable our model to operate in embodied AI contexts, we use *DROID* [29] with the improved calibration parameters provided in [27]. This dataset was captured at many locations around the world, and laboratories tend to have significantly different appearance, lighting, camera positions, and calibration quality. We also include a large collection of internally recorded bi-manual and single-arm tabletop robotics demonstrations, denoted *LBM*.
- **3D:** Because multi-view video is expensive to collect and therefore rather small in overall scale, we leverage single-view, posed videos of static scenes as an additional data source. Following [43, 68], we adopt *DL3DV-10K* [34] and *RealEstate-10K* [72]. We also include *ScanNet* [11], *TartanAir* [54], and *WildRGB-D* [59]. Because these environments are not dynamic, each frame can essentially be handled as if it were an independent camera, without any inherent temporal ordering. We randomly sample non-overlapping segments of 41 frames at training time, and treat them as two separate viewpoints.
- **Other:** This catch-all category covers all remaining multi-view video datasets, including *Kubric-4D* [50] and *Kubric-5D* [16] with synthetic multi-object interaction and physics, as well as *i.e.* *Ego-Exo4D* [15], depicting complex human activities in cluttered scenes.

In Figure 12, we provide additional examples of input and target camera poses of various episodes across training and evaluation sets to illustrate the diversity.

C.1. Kubric-5D

Kubric-5D is our newly introduced extension of Kubric-4D, with a new set of clips rendered with significantly more complex camera configuration and object placement. Compared to Kubric-4D, in which cameras are static with constant focal length, facing a small cluster of free-falling objects, *Kubric-5D* introduces dynamic cameras with varying focal lengths as well as varying object placement density, with the intent to enrich the dynamic information captured in the videos for the model to learn from. Specifically, we rerendered 1000 randomized scenes, each scene containing 16 cameras spawn at locations evenly distributed around the world center, and each camera’s trajectory type independently sampled; as for the focal length, 1/3 chance all 16 cameras in a scene share a preset value, 1/3 chance share a

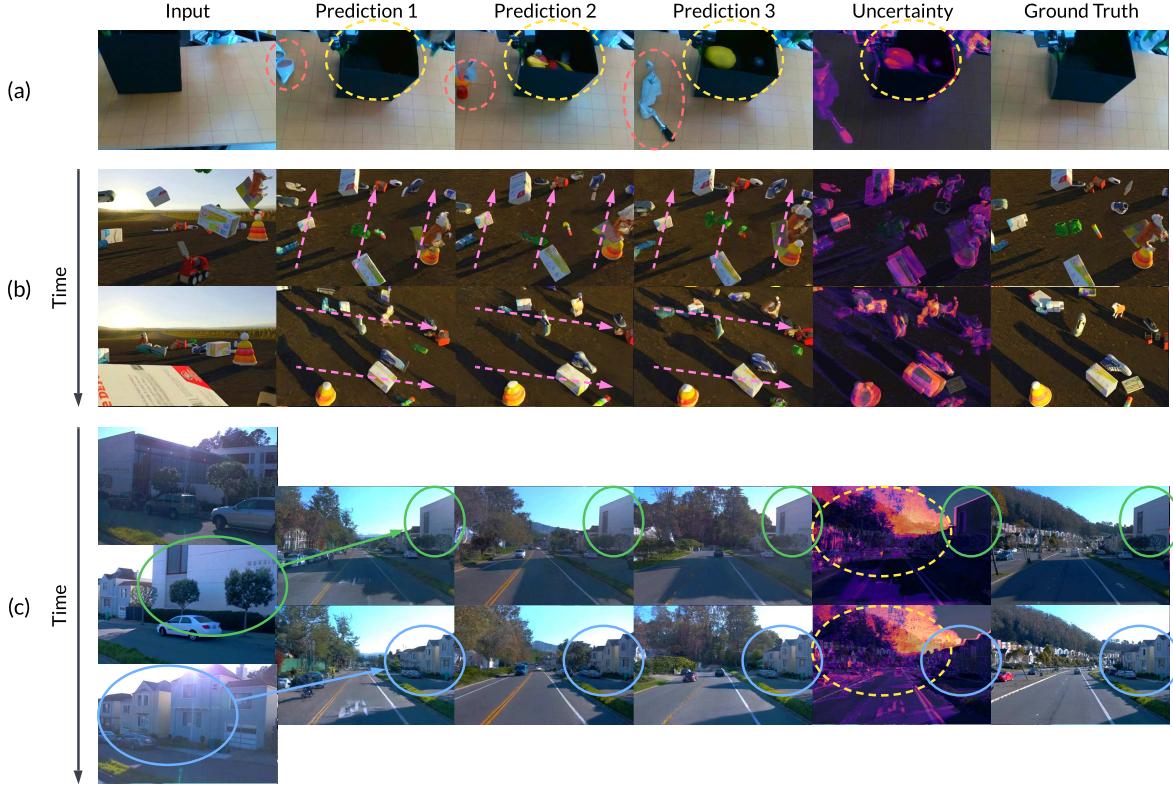


Figure 9. **Uncertainty analysis.** In (a), the model cannot see what is contained **inside the black bin** because the contents are occluded, and resorts to predicting fruit (since those objects are common in LBM), in addition to spawning **spurious objects** out-of-frame on the left. In (b), we mainly observe variations of object positions along the **input viewing direction** (overlaid with pink arrows for clarity), which presumably stems primarily from uncertainty in terms of implicit depth estimation that the model has to perform internally as part of the representation. In (c), only the front-right view is seen, which passes by **several buildings** that are reconstructed correctly in all samples (= front view). Meanwhile, the **left half** of these output videos has more diversity since it is never directly observed.

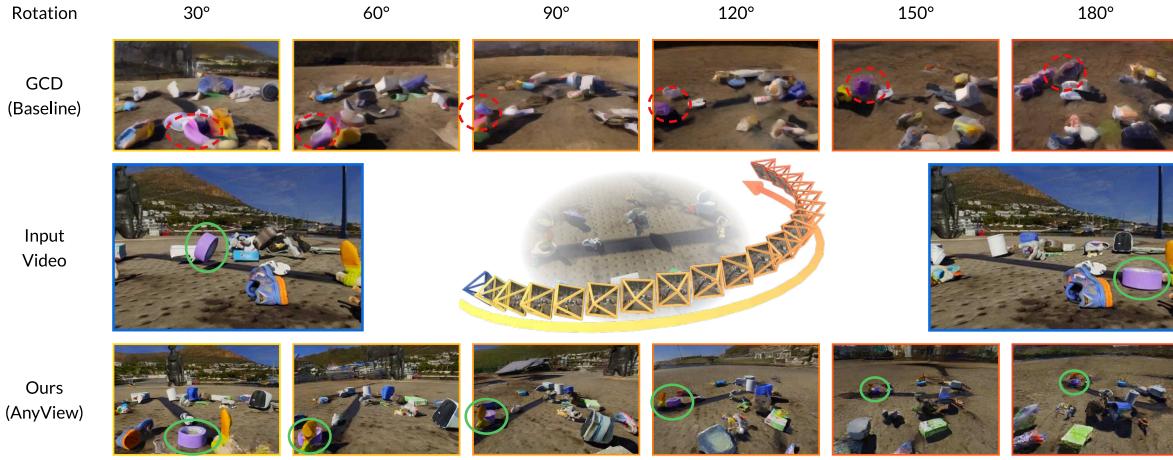


Figure 10. **Gradually increasing target azimuth.** As we increase the difficulty of the task by rotating the virtual camera over larger and larger angles away from the observed camera in this Kubric scene, GCD [50] produces garbled outputs where objects become **essentially unrecognizable**. In contrast, AnyView maintains **clear spatiotemporal correspondence** across dramatic viewpoint changes, demonstrating significantly enhanced 4D understanding over previous methods.

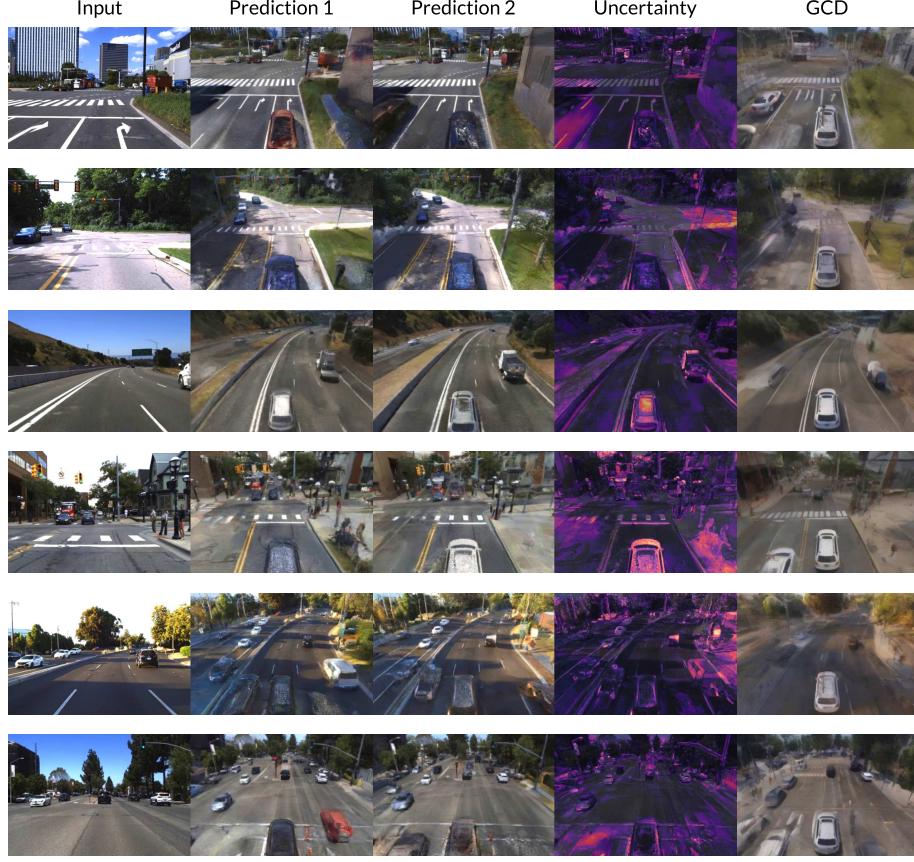


Figure 11. **Upward view synthesis on real-world driving scenarios.** We compare AnyView with GCD [50] on DDAD [17], which is a zero-shot dataset for both methods. AnyView generates much clearer predictions: almost every car that the model can see is reconstructed with high fidelity and accurate dynamics, whereas GCD often suffers from blurry artefacts, which worsen the further away one looks from the ego vehicle.

Dataset	S/R	Domain	Type	# Cameras	# Episodes	Resolution	Weight
DL3DV-10K [34]	Real	Indoor + Outdoor	3D	–	5,906	41 frames @ 384 × 208	6.3 %
DROID [29]	Real	Robotics	4D	2 (exo only)	29,712	29 frames @ 384 × 208	12.5 %
Ego-Exo4D [15]	Real	Human Activity	4D	4 – 5 (exo only)	2,489	41 frames @ 384 × 208	9.4 %
LBM	Sim + Real	Robotics	4D	2 (exo only)	53,886	41 frames @ 336 × 256	12.5 %
Kubric [16]	Sim	Multi-Object	4D	16 (exo only)	12,400	41 frames @ 384 × 256	15.6 %
Lyft [23]	Real	Driving	4D	6 (ego only)	296	41 frames @ 384 × 320	3.1 %
ParallelDomain [1]	Sim	Driving	4D	19 (16 exo + 3 ego)	7,352	41 frames @ 384 × 256	18.8 %
RealEstate-10K [72]	Real	Indoor + Outdoor	3D	–	34,968	41 frames @ 384 × 208	6.3 %
ScanNet [11]	Real	Indoor	3D	–	1,357	41 frames @ 384 × 288	3.1 %
TartanAir [54]	Sim	Indoor + Outdoor	3D	–	369	41 frames @ 384 × 288	3.1 %
Waymo [48]	Real	Driving	4D	5 (ego only)	798	41 frames @ 384 × {176, 256}	3.1 %
WildRGB-D [59]	Real	Single-Object	3D	–	23,002	41 frames @ 384 × 288	6.3 %

Table 4. **AnyView training datasets.** We use a weighted mixture of both static and dynamic data sources that combines multiple domains of interest. For multi-view video (4D) datasets, if there are more than two cameras, we randomly sample an input + ground truth pair for each training sample. For static (3D) datasets, with videos typically consisting of only one moving camera, we randomly sample subclips and treat them as different cameras for the purposes of training and evaluation.

randomly sampled value, and 1/3 chance each camera has an independently sampled value. Combining a geometry selection such as spiral, radial, line, lissajous,

etc., with the camera’s viewing direction, there are 16 different types of trajectories (including being static). The number of objects as well as spawn area are also ran-

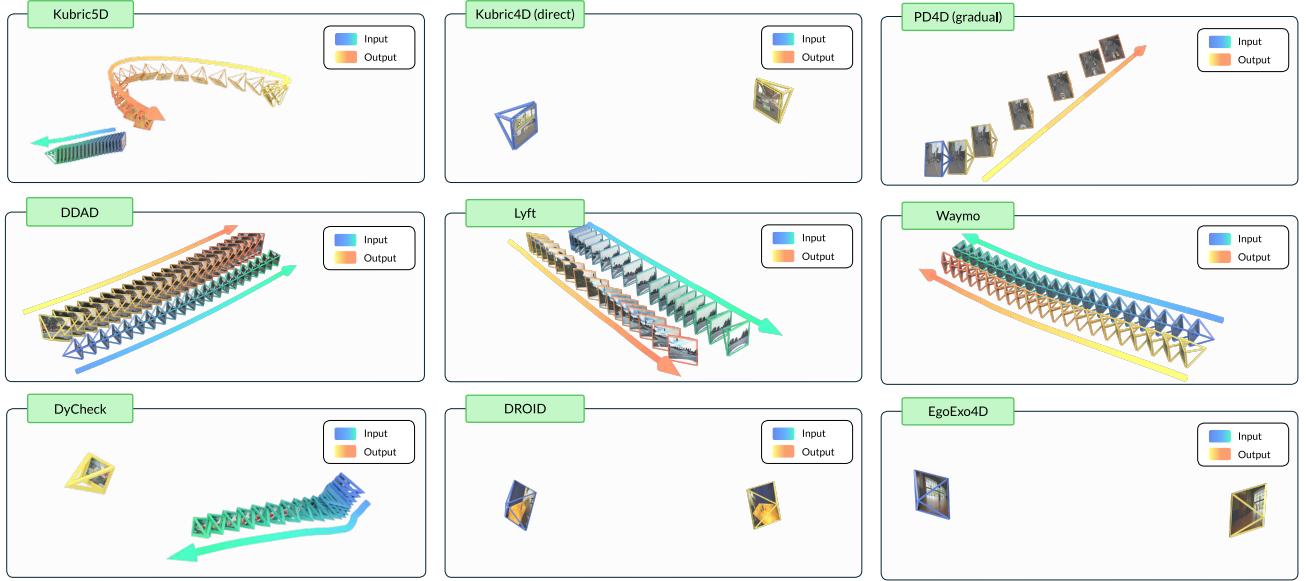


Figure 12. **Diversity of camera trajectories.** Samples of dataset camera trajectories illustrating the diversity of motion patterns used in our evaluation.

Method	Base Model	Training Datasets	Resolution	Input Cam.	Align Start	# DOF
GCD [50] (2024)	SVD-IB [6]	Kubric-4D, ParDom-4D	14 frames @ 384 × 256	Either	Either	3
TrajAttn [60] (2024)	SVD-IB [6]	MiraData	25 frames @ 1024 × 576	Flexible	Yes	$6 \cdot T$
GEN3C [43] (2025)	GEN3C-Cosmos-7B	Kubric-4D, DL3DV, RE-10K, Waymo OD	121 frames @ 1280 × 704	Moving	Yes	$6 \cdot T$
TrajCrafter [68] (2025)	CogVideoX-Fun-5B [64]	OpenVid-1M, DL3DV, RE-10K	49 frames @ 672 × 384	Flexible	Yes*	5
ReCamMaster [4] (2025)	Wan2.1 [52]	MultiCamVideo	81 frames @ 672 × 384	Flexible	Yes	< 1
InverseDVS [66] (2025)	CogVideoX-5B-I2V [64]	–	49 frames @ 720 × 480	Flexible	Flexible	$6 \cdot T$
CogNVS [9] (2025)	CogVideoX-5B-I2V [64]	SA-V, TAO, YT-VOS, DAVIS	49 frames @ 720 × 480	Moving	Flexible	6
Ours (AnyView)	Cosmos-2B [37]	See Table 4	[9, 41] frames @ 576 × [304, 384]	Flexible	Flexible	$6 \cdot T$

Table 5. **Description of baselines.** Some methods are self-supervised [9, 60, 68] and/or training-free [66], and hence do not require multi-view video datasets for training. *Input Cam.* refers to what kind of video a model can accept as input. *Align Start* specifies whether the output trajectory needs to start at the same initial frame, in which case we typically apply the smooth interpolation procedure. See Section E for more information. *TrajCrafter is trained with aligned start, but the official implementation does include limited support for non-aligned starting point inference.

domly sampled for each scene, covering the possibilities of denser/sparser clustering/scattering. All videos are rendered at 576×384 resolution with 24 FPS for 60 seconds, using the Kubric engine [16] and code adapted from [50].

D. Evaluation Datasets

Here, we describe the logic of which datasets and subsets are held out for evaluation purposes.

- **Driving:** The training sets for *Lyft* and *Waymo* are both recorded exclusively in the United States [23, 48]. We hold out *Argoverse*, also recorded in the USA [56] (albeit in mostly non-overlapping cities), because it has portrait videos as the front camera, which do not exist during training. We also hold out *DDAD*, because it contains videos recorded in Japan [17].

- **Robotics:** While episodes in LBM are recorded across multiple stations in both simulation and the real world, *DROID* [29] has more visual diversity. We decide to hold out all videos belonging to 2 out of 13 institutions (Gupta Lab, ILIAD) for zero-shot testing.

- **Human Activity:** One natural choice for this category is *Ego-Exo4D* [15], which has highly challenging, real-world scenes, often involving multiple humans, recorded by 4 to 5 inward-facing cameras. We hold out two *institutions* (FAIR, NUS), two *activities* (cpr, guitar), and three *institution-activity pairs* (basketball at Uniandes, piano at Indiana, soccer at UTokyo). Notably, *cpr at NUS* becomes the “most zero-shot” combination since both the activity and institution are entirely unseen. Since the cameras used to collect the dataset have noticeable distortion

tion, we implement a non-pinhole camera model to generate the actual viewing rays when given a grid, based on the official code examples that undistort the frames using coefficients stored in each sample. We further evaluate on videos from the eight exocentric cameras of the *AssemblyHands* [38] dataset, a subset of *Assembly101* [45] that has calibrated camera intrinsics and extrinsics. The dataset records dexterous hand-object interactions during the assembly and disassembly of pull-apart toys, providing a challenging zero-shot test setting for AnyView.

E. Baselines

Here, we outline how each baseline was adapted to AnyViewBench. In each case, when a method predicts *fewer* frames than the evaluation episode, we run the model multiple times in a sliding window fashion until the full video is covered, and average metrics such that each frame is used exactly once. In the opposite scenario, *i.e.* when a method predicts *more* frames than necessary, we simply discard the superfluous ones.

We provide the evaluated methods with ground truth camera pose and intrinsics, and when a method needs depth we use DepthAnythingV2 [63] to calculate metric depths maps since the ground truth pose we use are in metric space.

Some methods are trained to operate with smooth camera trajectories, and their performance degrades when there is minimal overlap between the target and input trajectories in the beginning of the videos. However, many trajectories in AnyViewBench exhibit precisely such limited overlap. To address this, we use the estimated depth to smoothly interpolate between the input view and the first target view, freezing the first frame for a short while until the target pose is reached, then concatenate these interpolated frames with the actual input sequence.

- **Generative Camera Dolly (GCD) [50]:** This model only supports inference with 14 frames at a time (both in terms of input and output video), and with 3 degrees of freedom. It assumes a spherical coordinate system (ϕ, θ, r) , where the camera controls provided to the network are the relative azimuth angle $\Delta\phi$, relative elevation angle $\Delta\theta$, and relative radius Δr . The input and target viewpoints always aim at the center of the scene. To reduce the $6 \cdot T$ -DOF AnyViewBench camera trajectories into the 3-DOF conditioning space of GCD, information loss is unavoidable, so we apply the following approximate projection:

1. Take the forward-looking vector $f = (f_x, f_y, f_z)$ (= third column of the extrinsics matrix) and translation vector $t = (t_x, t_y, t_z)$ (= last column of the extrinsics matrix) of the camera pose of each viewpoint of either the middle or last frame (depending on the dataset) of the video.
2. Measure the azimuth angle of each vector: $\phi =$

$\arctan\left(\frac{f_y}{f_x}\right)$; the difference between both values is then $\Delta\phi$.

3. Measure the elevation angle of each vector: $\theta = -\arctan\left(\frac{f_z}{\sqrt{f_x^2 + f_y^2}}\right)$; the difference between both values is then $\Delta\theta$.
4. Measure the Euclidean distance from each camera origin to the scene origin: $r = \sqrt{t_x^2 + t_y^2 + t_z^2}$; the difference between both values is then Δr .

- **Trajectory Attention [60]:** TrajectoryAttention takes a variable number of input image frames at a resolution of 1024×576 . Given N input images, we provide the N warped images from the target views along with the first image from the source view ($N+1$ images in total). Since our trajectories are represented in metric space, we opted to use the metric version of DepthAnythingV2, unlike the non-metric model used in the original implementation. We also modified the original warping code, which only supported transformations around the source view, so that it can handle arbitrary trajectories.

- **GEN3C [43]:** GEN3C supports number of frames in $120 * N + 1$ pattern; we choose 121 as it is enough to cover the length of clips in all evaluated datasets. To meet the length requirement, each input video is padded to 121 frames using the last frame, and metrics are only computed on the original leading frames from the output. Following the official inference code, the videos are first resized and predicted in 1280×704 , and we resize them back to the original resolution for metrics calculation. The original implementation requires per-frame camera pose, intrinsics, and depth map estimated by choice of SLAM packages (VIPE [26] recommended) for each video; while this is designed for arbitrary videos without 3D information, it prevents us from specifying desired camera poses and intrinsics for fair comparison with the ground truths. Therefore, we instead feed the pipeline ground truth camera poses, intrinsics, and depths maps estimated by DepthAnythingV2 as mentioned in the beginning of section. It is worth noting that VIPE’s estimated depth cannot be used alone in this case, as its scale is coupled with the estimated pose and intrinsics instead of ground truth ones.

- **TrajectoryCrafter [68]:** TrajCrafter supports 49-frame clips at 672×384 . The input camera is flexible. The original implementation relies on a parameterized trajectory representation (θ, ϕ, r, x, y) for spherical camera motion and computes geometric warping using depth estimated by DepthCrafter [24]. While suitable for smooth parametric trajectories, this approach has limited support for arbitrary real-world camera transformations, such as those found in our benchmark. To address this limitation, we modified the inference implementation to load pre-

computed re-projected RGB frames, bypassing the original depth estimation and re-projection steps. We apply the depth warping interpolation procedure as described above. Binary masks are automatically computed by thresholding black pixels to identify invalid re-projection regions. The rest of the implementation is left unchanged.

- **CogNVS [9]:** Similarly to TrajectoryCrafter, CogNVS supports 49-frame sequences at a resolution of 720×480 . We do not perform test-time optimization and instead run the model in a zero-shot manner. CogNVS can be combined with any depth reconstruction approach, allowing improved view synthesis through better geometric reconstruction. To ensure consistency with other baselines that rely on off-the-shelf depth estimators, we use monocular depth estimated by DepthAnythingV2. We apply the depth warping interpolation procedure as described above, matching the required 49-frame length.

We summarize the training sets and some properties of each baseline in 5. Here, “# DOF” stands for (continuous) degrees of freedom, denoting the dimensionality of the space of trajectories each model was trained with (ignoring intrinsics), and is thus linked to its *effective* camera pose controllability at inference time. < 1 means that only a finite list of possible canonical trajectories are supported. The “Input Cam.” options mean:

- Moving: The method expects the camera trajectory of the input video to move, *e.g.* for depth estimation to work well.
- Flexible: The same model can support either static pose or dynamic pose input videos.
- Either: Separate models exist for input videos with fixed or moving poses over time.

The “Align Start” options mean:

- Yes: The first target camera pose must be spatially very close to the first input camera pose (typically linked to narrow DVS).
- Flexible: The same model can support both narrow and extreme DVS.
- Either: Separate models exist for both settings.