ViDAR: Video Diffusion-Aware 4D Reconstruction From Monocular Inputs

Michal Nazarczuk[1*] **Sibi Catley-Chandar**[1,2*] **Thomas Tanay**[1]
Zhensong Zhang[1] **Gregory Slabaugh**[2] **Eduardo Pérez-Pellitero**[1]

[1] Huawei Noah's Ark Lab [2] Queen Mary University of London

Figure 1: ViDAR provides a novel framework for Monocular Novel View Synthesis utilising a diffusion-aware reconstruction framework.

Abstract

Dynamic Novel View Synthesis aims to generate photorealistic views of moving subjects from arbitrary viewpoints. This task is particularly challenging when relying on monocular video, where disentangling structure from motion is ill-posed and supervision is scarce. We introduce Video Diffusion-Aware Reconstruction (ViDAR), a novel 4D reconstruction framework that leverages personalised image diffusion models to synthesise pseudo multi-view supervision signals for training a Gaussian splatting representation. By conditioning on scene-specific features, ViDAR recovers fine-grained appearance details while mitigating artefacts introduced by monocular ambiguity. To address the spatio-temporal inconsistency of diffusion-based supervision, we propose a diffusion-aware loss function and a camera pose optimisation strategy that aligns synthetic views with the underlying scene geometry. Experiments on DyCheck, a challenging benchmark with extreme viewpoint variation, show that ViDAR outperforms all state-of-the-art baselines in visual quality and geometric consistency. We further highlight ViDAR's strong improvement over baselines on dynamic regions and provide a new benchmark to compare performance in reconstructing motion-rich parts of the scene. Project page: `https://vidar-4d.github.io/`.

[*] equal contribution

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

1 Introduction

4D reconstruction from monocular inputs is a challenging problem where the goal is to recover a 3D representation of a dynamic scene. It is increasingly important for modelling, comprehending, and interacting with the physical world and supports a wide range of downstream applications, ranging from augmented reality to generating data for training robust AI models [57].

Casually captured monocular videos are ubiquitous, however reconstructing 3D structure from them remains an inherently ill-posed problem. Static regions of the scene can typically be reconstructed well due to effective multi-view capture [5]. However for dynamic regions, depth information is not directly observable from a single viewpoint; in other words, it is difficult to disentangle the motion of the camera from motion within the scene. To mitigate this ambiguity many existing approaches impose strong regularisation [7; 30; 58] in the form of geometric assumptions, such as the object's rigidity, that constrains the dynamics of the scene. Others [13; 25; 46; 65; 66] leverage learned priors, particularly those derived from large-scale models (*e.g*. monocular depth), to guide the reconstruction. While regularization based methods [13; 46] achieve geometrically compact scene representations, they often fall short in rendering high-quality, photorealistic appearances. Conversely recent generative approaches utilise powerful diffusion models to achieve higher visual quality in tasks such as single image to 3D [21; 23; 24; 53; 54; 67] and monocular reconstruction [51] but struggle to maintain spatio-temporal coherence, limiting their applicability in scenarios that demand accurate spatial reconstruction and temporal consistency, particularly in dynamic, real-world settings.

To tackle these challenges, we present Video Diffusion-Aware 4D Reconstruction (ViDAR), a monocular video reconstruction approach that leverages diffusion models as powerful appearance priors through a novel diffusion-aware reconstruction framework, which allows for improving visual fidelity without the loss of spatio-temporal consistency. We first train a monocular reconstruction baseline and generate a set of typically degraded multi-view images by sampling diverse camera poses and rendering the novel viewpoints. We then adopt a DreamBooth-style personalisation strategy [37], and tailor a pretrained diffusion model to the input video, which we use as a generative enhancer to inject rich visual information back into the degraded renders. This effectively generates a set of high-fidelity pseudo-multi-view observations for our scene, although due to the nature of the diffusion process, the resulting images are not necessarily spatially consistent. We observe that naively using these views as supervision leads to reconstructions degraded by artefacts and geometric inconsistencies. To mitigate this, we propose a method of diffusion-aware reconstruction, which selectively applies diffusion-based guidance to dynamic regions of the scene while jointly optimising the camera poses associated with the diffused views.

To the best of our knowledge, ViDAR is the first approach to incorporate a diffusion prior into monocular video reconstruction in a geometrically consistent manner. We demonstrate substantially improved qualitative and quantitative results compared to existing techniques (see Tabs. 1, 2, Figs. 1, 4), highlighting the effectiveness of diffusion-guided supervision when integrated with a reconstruction pipeline that accounts for geometric consistency.

We summarise our contributions as follows:

1. A personalised diffusion enhancement strategy that improves appearance quality by refining newly sampled renderings using a DreamBooth-adapted model.
2. A diffusion-aware reconstruction framework that combines dynamic-region-focused diffusion guidance with joint optimisation of the sampled camera poses for geometrically consistent reconstruction.
3. An extensive experimental evaluation, including both quantitative and qualitative comparisons with prior work, the introduction of a dynamic-region specific benchmark, as well as ablation studies isolating the impact of each component.

2 Related work

4D reconstruction Advances in novel view synthesis include the introduction of two seminal reconstruction paradigms, namely Neural Radiance Fields (NeRF) [29] and 3D Gaussian Splatting (3DGS) [11]. These developments in static scene reconstruction were quickly followed by several works on dynamic content. NeRF-based methods for video reconstruction include D-NeRF [35], StreamRF [15], HexPlane [2], K-Planes [3], Tensor4D [38], MixVoxels [45]. Similarly, Gaussian

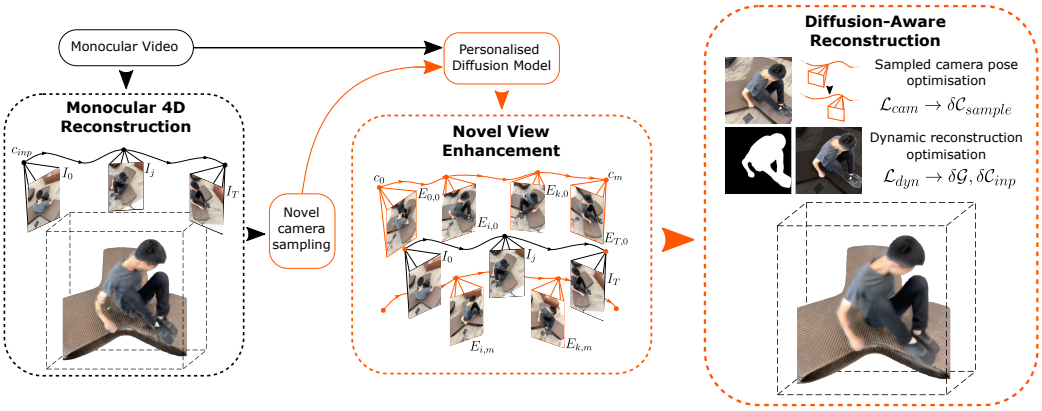


Figure 2: A high-level overview of ViDAR. The input video is used to create a 4D reconstruction with a monocular approach. Further, novel camera views are sampled and enhanced with a personalised diffusion model for each scene. This constitutes a set of pseudo-multi-view supervision examples. Finally, our approach optimises the 4D representation with the use of original video and new multi-view cues, in a diffusion-aware manner.

Splatting developments enabled research on dynamic novel view synthesis. Multi-view videos were reconstructed by: GaussianFlow [22], 4DGS [49], STG [18], SWinGS [39], Ex4DGS [12].

Monocular reconstruction The task of 4D monocular video reconstruction can be seen as a special case of 4D reconstruction under substantially more challenging conditions. This is due to the problem often being ill-posed: many of the target object surfaces may be seen only from one viewpoint throughout the video. Notably, among NeRF-based approaches, NSFF [16] proposes a time varying flow field, whereas Nerfies [31], HyperNeRF [32], DyCheck (T-NeRF) [5], DyBluRF [1], RoDynRF [26], CTNeRF [28] use a canonical representation with a time-dependent deformation. DynIBaR [17] uses Image Based Rendering for reconstruction. With Gaussian Splatting advancements, Dynamic 3D Gaussians [27] learn explicit motion of every Gaussian, whereas 4DGS [49], Deformable 3DGS [58], SC-GS [7] use a deformation field for transformation from canonical space. SplineGS [30] constrains the motion of Gaussians to splines to ensure temporal smoothness. DynPoint [65] and MotionGS [66] use an optical flow estimator for additional supervision. PGDVS [64] and BTimer [19] propose a transformer-based approach for generalisable reconstruction. Dynamic Gaussian Marbles [42] adopt a divide-and-conquer strategy to merge sets of Gaussians and create long trajectories, and restrict representation to isotropic Gaussians. MoDGS [25] improves the supervision from depth priors. D-NPC [9] proposes the use of neural implicit point cloud as the representation for monocular reconstruction. MoSca [13] and Shape of Motion [46] both utilise priors from pretrained foundational models (depth, optical flow, 2D tracking). Similarly, they both reconstruct static and dynamic content separately, and describe the motion of the Gaussians with lower dimensionality basis functions.

Diffusion enhanced reconstruction Several recent approaches explore the use of diffusion models to guide the reconstruction. ReconFusion [52] trains a diffusion model on a set of object images, and uses it to score the quality of sparse reconstruction, guiding it with RGB loss. DpDy [44] uses Score Distillation Sampling (SDS) [34] to supervise reconstruction with the use of image and depth diffusion model. CAT4D [51], uses a video-diffusion model to generate additional static cameras for the input video, followed by the reconstruction process. Difix3D+ [50] proposes a generalisable enhancement diffusion model to improve reconstruction quality. MVGD [6] proposes a direct rendering of novel views and depth as a conditional generative task. Other diffusion-based approaches include text or a single image to 3D generation [14; 20; 21; 23; 24; 40; 43; 48; 53; 54; 55; 59; 61; 62; 67]. Notably, our approach uses a monocular video as an input, and uses a personalised diffusion model along with our diffusion-aware reconstruction for accurate geometry modelling.

3 Method

Our method incorporates several stages which can be seen in Figure 2. Firstly, we use a monocular reconstruction baseline to obtain a 4D representation of the scene (Sec. 3.1), and generate a set of

degraded multi-view renders from sampled novel camera poses. Next, we personalise a diffusion model using the input video (Sec. 3.2), which is used to enhance the degraded renders (Sec. 3.2.1). Finally, we use the new set of enhanced pseudo-multi-view images to supervise and refine the 4D representation of the scene (Sec. 3.3), in a diffusion-aware manner.

3.1 Monocular Reconstruction

Given a casual monocular video of a dynamic scene with T frames $\mathcal{I} = [I_1, I_2, \dots I_T]$, we perform the initial reconstruction of the scene using an off-the-shelf 4D monocular reconstruction method, specifically, we use MoSca [13] in our implementation. The method reconstructs two sets of Gaussians for the given scene, namely static Gaussians $\mathcal{G}_s$, and dynamic Gaussians $\mathcal{G}_d$, that together create the scene representation: $\mathcal{G} = \mathcal{G}_d \cup \mathcal{G}_s$. MoSca leverages several priors in the reconstruction process: depth, optical flow, 2D tracking. Firstly, the optical flow is used to estimate the epipolar error and to determine the likelihood of image regions belonging to dynamic or static content. This is followed by the joint reconstruction of the static part of the scene $\mathcal{G}_s$ and fine-tuning of the input camera pose c_{inp}. With that, a *scaffold*, a low dimensionality motion representation, is built through lifting 2D tracklets belonging to dynamic regions into 3D using depth information. Finally, a photometric reconstruction is performed to optimise the scene $\mathcal{G}$, enabling rendering of novel views R of the scene.

3.1.1 Track Anything Gaussian Classification

We note that the epipolar error analysis introduced by MoSca for classification of dynamic parts of the image leads to occurrences of floater artefacts due to the inclusion of background among dynamic Gaussians. This may not be reflected heavily in quantitative performance, but leads to a decrease in the quality of generated pseudo-multi-view samples (Sec. 3.2.1). To improve the constraint on dynamic Gaussians' locations, we use dynamic masks D_t obtained from Track Anything [56] to reconstruct the static part of the scene $\mathcal{G}_s$ and generate motion scaffolds (as in MoSca [13]).

3.2 Diffusion Enhancement

We utilise a Stable Diffusion [36] model, specifically the pretrained Stable Diffusion XL (SDXL) [33] to improve the quality of rendered images and guide the reconstruction process. Following the observations of ReconFusion [52], we decide to use a multistep denoising process, in contrast to Score Distillation Sampling [34]. Conversely, given a sampled image $R_{m,t}$ from camera c_m at the time t, we follow a standard text-to-image [36] process and encode the image into latent space: $x_0 = \mathcal{E}(R_{m,t})$. Further, instead of generating the noisy latent for image generation, we introduce k steps of noise into the image-sourced latent $x_0 \rightarrow x_k$ using the original noise scheduler (here, Discrete Euler [10]). We then follow the denoising process for k steps to achieve a denoised latent $\hat{x}_0$ which is then decoded to an enhanced version of the input image: $E_{m,t} = \mathcal{D}(\hat{x}_0)$.

Personalisation Similarly to some of the recent reconstruction approaches, *e.g.* Wang et al. [44], we apply the Dreambooth [37] fine-tuning approach to the SDXL model. To this end, we treat an input video $\mathcal{I}$ as a collection of images and fine-tune the diffusion model for a given scene such that a specific text token triggers the model to follow the appearance of the scene.

3.2.1 View Sampling and Rendering Enhancement

Given the scene-personalised diffusion model, we utilise the previously trained monocular reconstruction to generate a set of pseudo-multi-view ground truth images. Firstly, we sample M sets of images for each timestep $t \in [0, T]$, effectively adding M new cameras with parameters c_m where $m \in [0, M]$ and $c_m \in \mathcal{C}_{sample}$, where $\mathcal{C}_{sample}$ constitutes a set of new camera trajectories. To this end, we select two existing views (as a camera position and rotation): a random one, and a challenging view (with the furthest distance from the mean), and sample a new view as their weighted linear combination. To introduce variety in the difficulty of the sampled views, we gradually increase the blending weight of the views towards the most challenging ones from the input trajectory. Simultaneously, we introduce noise of an increasing amplitude to the new camera poses.

Thereafter, we use our trained monocular reconstruction to render a set of M new camera views $R_{m,t}$ for each timestep. Further, we use our personalised diffusion model to enhance the rendered images $R_{m,t} \rightarrow E_{m,t}$. This constitutes a new set of supervision images in a multi-view setting. We have

chosen to generate a whole multi-view dataset in a single step instead of performing the enhancement on-the-fly. This enables the samples to be reused and reduces the computational demands (especially on GPU memory).

3.3 Diffusion-Aware Reconstruction

We use our generated dataset $\{E_{m,t}\}$ as additional supervision to re-train our 4D monocular reconstruction method to predict a higher quality output $\hat{I}_{m,t}$. However using these sampled views for training is challenging. The outputs $E_{m,t}$ of our personalised video diffusion models are high-fidelity and also preserve structure and coarse geometry, but due to the nature of the diffusion process and random noise schedule, they are not spatio-temporally consistent at the level of fine-grained detail and texture. This manifests as flickering and shifts of textures between consecutive frames. In some cases, coarse geometry may also be hallucinated, e.g. in novel viewpoints not seen during training. If we naively used these outputs to supervise monocular 4D reconstruction, the lack of spatio-temporal consistency in the training data would cause the model to either converge to a mean radiance value and cause blurry renderings, or to overfit to individual frames and learn a temporally inconsistent reconstruction. We propose the following mechanisms to overcome these challenges.

3.3.1 Dynamic Reconstruction

While dynamic regions of a scene are under-observed, static regions may be captured from multiple viewpoints across time, effectively creating multi-view supervision. Hence supervision is unnecessary in static regions and in fact could cause the quality to decrease, particularly if spatially inconsistent. We compute a mask of the dynamic regions of the scene $D_{m,t}$ using Track Anything [56], and apply this mask to our data to mask out the static regions $E^{dyn}_{m,t} = E_{m,t} \odot D_{m,t}$, where $\odot$ denotes element-wise multiplication, and also to our predicted output $\hat{I}^{dyn}_{m,t} = \hat{I}_{m,t} \odot D_{m,t}$. This ensures that only the dynamic regions of the scene are supervised by our generated data, which reduces the convergence to the mean effect in the static reconstruction and reduces floaters. For dynamic regions, we introduce a perceptual loss [63] to encourage our reconstruction to be texturally rich and reduce blur caused by training on spatially misaligned pseudo-GTs. During training we compute the loss as $\mathcal{L}_{dyn} = |E^{dyn}_{m,t} - \hat{I}^{dyn}_{m,t}|_1 + \lambda_p|E^{dyn}_{m,t} - \hat{I}^{dyn}_{m,t}|_{vgg} + \lambda_s|E^{dyn}_{m,t} - \hat{I}^{dyn}_{m,t}|_{ssim}$, where $|\cdot|_1$ is the L1 loss, $|\cdot|_{vgg}$ is the perceptual loss using a pretrained VGG network [41], $|\cdot|_{ssim}$ is the SSIM [47] loss and λ_p and λ_s are hyperparameters set to 0.1. The dynamic loss, $\mathcal{L}_{dyn}$, is applied in addition to the default losses from the monocular reconstruction method, and is backpropagated to update $\mathcal{G}$ and $\mathcal{C}_{inp}$.

3.3.2 Sampled Camera Pose Optimisation

Camera poses of casually captured monocular videos are typically noisy due to the difficulty of disentangling scene motion from camera motion, thus the need to optimise $\mathcal{C}_{inp}$ in many monocular reconstruction methods [13; 46]. Our sampled camera poses $\mathcal{C}_{sample}$ are interpolated from $\mathcal{C}_{inp}$ and so are also noisy. As our pseudo-GTs corresponding to $\mathcal{C}_{sample}$ are not always spatially consistent, it is even more difficult to disentangle scene motion from camera motion. To compensate for this, it is necessary to optimise our sampled camera poses during training to ensure the pseudo-GTs are aligned with the underlying scene geometry. However unlike dynamic reconstruction (Sec. 3.3.1) where we only use the dynamic masked region for supervision, we use the entire image $E_{t,m}$ as supervision for sampled camera pose optimisation. Despite fine-grained textural flickering, the coarse structure present in static regions provides a more consistent supervision signal for localisation than using only dynamic regions. We compute the loss as $\mathcal{L}_{cam} = |E_{m,t} - \hat{I}_{m,t}|_1 + \lambda_p|E_{m,t} - \hat{I}_{m,t}|_{vgg} + \lambda_s|E_{m,t} - \hat{I}_{m,t}|_{ssim}$. The camera loss, $\mathcal{L}_{cam}$, is backpropagated separately to other losses and only updates $\mathcal{C}_{sample}$.

4 Results

4.1 Datasets

We evaluate the performance of ViDAR on the DyCheck dataset [5]. DyCheck was introduced as a real world benchmark for evaluating monocular to 4D methods and is extremely challenging: the

Table 1: Quantitative results on co-visibility masked regions of scenes from the DyCheck (iPhone) dataset. Best, second and third results are highlighted in red, orange and yellow respectively. SoM-5 is full-res with wheel and space-out excluded.

	Method	PSNR-m↑	SSIM-m↑	LPIPS-m↓
Half-Res	T-NeRF [5]	16.96	0.5772	0.3789
	NSFF [16]	15.46	0.5510	0.3960
	Nerfies [31]	16.45	0.5699	0.3389
	HyperNeRF [32]	16.81	0.5693	0.3319
	4DGS [49]	13.64	-	0.4280
	PGDVS [64]	15.88	0.5480	0.3400
	DynPoint [65]	16.89	0.5730	-
	DyBluRF [1]	17.37	0.5910	0.3730
	D-NPC [9]	16.41	0.5820	0.3190
	RoDynRF [26]	17.10	0.5340	0.5170
	Gaussian Marbles [42]	16.02	0.5416	0.3398
	SoM [46]	18.62	0.6820	0.2382
	MoSca [13]	19.32	0.7060	0.2640
	Ours	19.69	0.7126	0.2231
Full-Res	Gaussian Marbles [42]	15.84	0.5434	0.5681
	SoM [46]	17.98	0.6422	0.3718
	MoSca [13]	18.44	0.6560	0.4193
	Ours	19.00	0.6672	0.3623
SoM-5	T-NeRF [5]	15.60	0.5500	0.5500
	HyperNeRF [32]	15.99	0.5900	0.5100
	DynIBaR [17]	13.41	0.4800	0.5500
	Gaussian Marbles [42]	16.03	0.5425	0.5807
	SoM [46]	16.72	0.6300	0.4500
	CAT4D [51]	17.39	0.6070	0.3410
	MoSca [13]	18.34	0.6636	0.4321
	Ours	18.76	0.6751	0.3774

Table 2: Quantitative results on dynamic regions of scenes from the DyCheck (iPhone) dataset. Best, second and third results are highlighted in red, orange and yellow respectively. SoM-5 is full-res with wheel and space-out excluded.

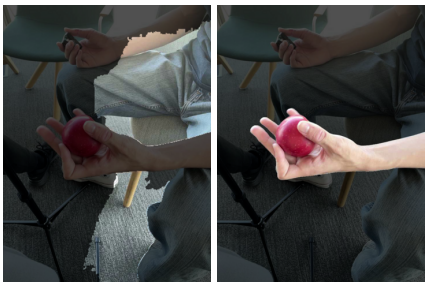
	Method	PSNR-D↑	SSIM-D↑	LPIPS-D↓
Half-Res	T-NeRF [5]	13.86	0.8546	0.3491
	Nerfies [31]	12.89	0.8425	0.3811
	HyperNeRF [32]	13.27	0.8484	0.3558
	Gaussian Marbles [42]	9.99	0.8175	0.3926
	SoM [46]	14.80	0.8582	0.3008
	MoSca [13]	15.63	0.8755	0.2904
	Ours	16.46	0.8850	0.2793
Full-Res	Gaussian Marbles [42]	12.75	0.8607	0.5058
	SoM [46]	14.82	0.8709	0.4347
	MoSca [13]	15.39	0.8821	0.4413
	Ours	16.32	0.8893	0.3921
SoM-5	Gaussian Marbles [42]	13.66	0.8658	0.4919
	SoM [46]	12.50	0.8648	0.4890
	MoSca [13]	15.83	0.8872	0.4404
	Ours	16.69	0.8941	0.3778

Table 3: Quantitative results on NVIDIA dataset, full image and dynamic region included. Best, second and third results are highlighted in red, orange and yellow respectively.

Method	PSNR↑	SSIM↑	LPIPS↓	PSNR-D↑	SSIM-D↑	LPIPS-D↓
DynNeRF [4]	26.10	0.8370	0.0798	19.90	0.9536	0.1268
RoDynRF [26]	25.89	0.8538	0.0648	19.59	0.9544	0.1280
NSFF [16]	24.33	0.7511	0.1939	19.51	0.9522	0.1928
MoSca [13]	26.72	0.8507	0.0708	21.16	0.9405	0.1238
Ours	27.20	0.8610	0.0613	21.35	0.9613	0.1059

Table 4: Intersection of co-visibility mask with dynamic regions with respect to co-visibility mask area

Scene	Dyn/Co-vis Intersection (%)
apple	4.42
block	27.46
paper-windmill	3.58
space-out	20.63
spin	19.76
teddy	81.33
wheel	24.65
mean	25.97



Co-visibility mask Dynamic mask

Figure 3: An example of co-visibility and dynamic mask comparison.

test views are far away from training views, camera poses are often inaccurate, depths are noisy and training views have issues such as overexposure and autofocus. The dataset consists of 14 casually captured scenes, 7 of which have no ground truth test views and are used for qualitative evaluation only and 7 with test views available. Due to the difficulty of obtaining accurate camera poses for all scenes, some methods choose to quantitatively evaluate on only 5 of the available 7 scenes and discard 'space-out' and 'wheel'. To our knowledge, this is currently the widely used benchmark which is the most appropriate for evaluating our method. As described in DyCheck [5], other datasets such as Nerfies [31] , HyperNeRF [32] and NSFF [16] suffer from teleporting cameras which makes them effectively multi-view. We quantitatively and qualitatively evaluate our method and other state-of-the-art baselines across all 14 scenes. In addition, we provide an evaluation of ViDAR on the NVIDIA dataset [60]. Notably, it is a forward-facing capture with small-baseline static cameras, thus significantly easier than DyCheck.

4.2 Metrics

Following previous works [5; 13], we compute PSNR, SSIM and LPIPS on the co-visibility masked regions of the test views, which we denote with an *-m* addendum to each metric. We compute metrics at both half-resolution and full-resolution, and following [46], we also report results on a subset of 5 scenes which we label SoM-5. For the NVIDIA dataset we follow the setup of MoSca proposed in RoDynRF [26].

4.2.1 Limitations of Metrics and A New Benchmark

We note that the static regions of a scene are often observed from several viewpoints across different time steps in the captured monocular video. This effectively provides multi-view supervision for these regions, and although we are interested in reconstructing the entire scene which includes the static regions, the dynamic regions are arguably the area of most interest and also the most under-observed. In order to better evaluate performance in the dynamic regions of the scene, we compute a set of dynamic masks for each scene using Track Anything [56]. We compute the intersection between the co-visibility masks and the dynamic regions of the scene and present results in Table 4. We find that on average only 26% of the co-visibility masked pixels correspond to the dynamic region. Some scenes such as apple and paper-windmill have an intersection as low as 4%. We show an example of this in Figure 3. The co-visibility masked metrics are heavily weighted towards the static regions of

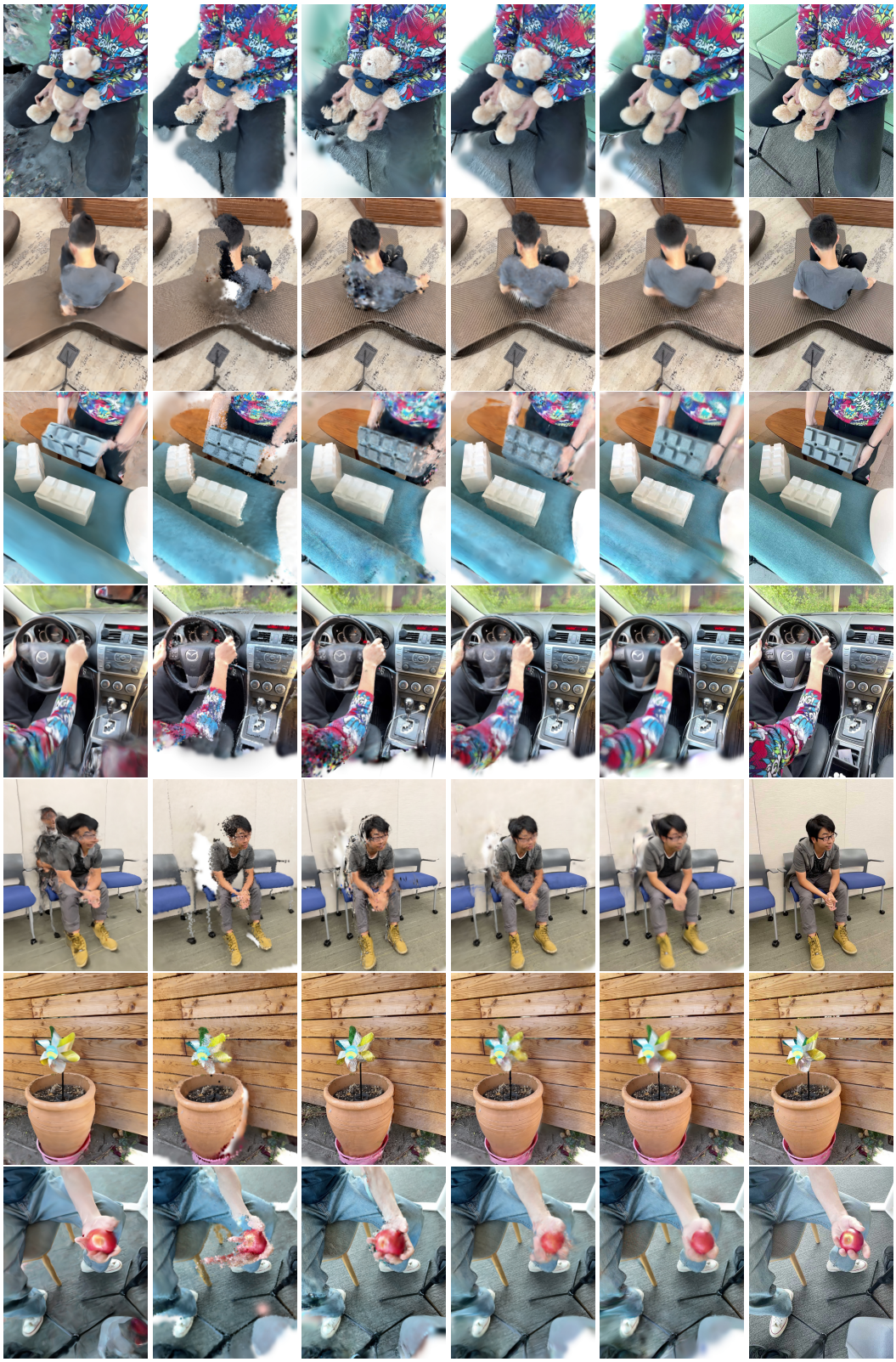


Figure 4: Qualitative evaluation of our method against benchmark methods on the DyCheck test set.

Figure 5: Qualitative evaluation of our method against benchmark methods on the NVIDIA test set.

Table 5: Quantitative results of an ablation study of the components of ViDAR.

Method	PSNR-m	SSIM-m	LPIPS-m
Ours	19.00	0.6672	0.3623
W/o Tracking Based Gaussian Classification (TGS)	18.88	0.6651	0.3693
W/o Sampled Camera Optimisation (SO)	18.39	0.6514	0.4040
W/o Dynamic Reconstruction (DR)	18.93	0.6274	0.4497
W/o SO + DR + TGS	18.46	0.6075	0.4656
W/o diffusion (TGS only)	18.65	0.6596	0.4075
MoSca	18.44	0.6560	0.4193

the scene. Although this is useful for evaluating overall reconstruction performance, it underweights the reconstruction performance of methods in the most difficult dynamic regions. We provide a complementary new benchmark for the evaluation of monocular to 4D reconstruction methods, where our computed dynamic masks can be used in place of the commonly used co-visibility masks. We use these masks to compute the PSNR, SSIM and LPIPS, which we denote with a *-D* addendum, for a range of baseline methods in Table 2.

4.3 Evaluation

Baselines We compare against a wide range of baselines, including a number of recent state-of-the-art methods such as MoSca [13], CAT4D [51], Shape Of Motion [46], Dynamic Gaussian Marbles [42] and 4DGS [49], which are based upon Gaussian Splatting [11]. We also compare against NeRF-based approaches T-NeRF [5], Nerfies [31], HyperNeRF [32], DyBluRF [1] and RoDynRF [26], neural point clouds approaches DynPoint [65] and D-NPC [9], generalized pre-trained transformer PGDVS [64], neural scene flow NSFF [16] and volumetric image-based rendering DynIBaR [17].

Quantitative and Qualitative Evaluation We present quantitative results of our method in Tables 1 and 2. Our method outperforms all state-of-the-art baselines in PSNR and SSIM and all but one in LPIPS, across all settings and resolutions. We typically improve PSNR by a large margin, achieving a minimum of 1dB improvement over all methods, except for MoSca where we average 0.94dB and 0.56dB higher in dynamic and co-visibility masked regions respectively. This indicates our method particularly improves dynamic region reconstruction. We note that CAT4D achieves

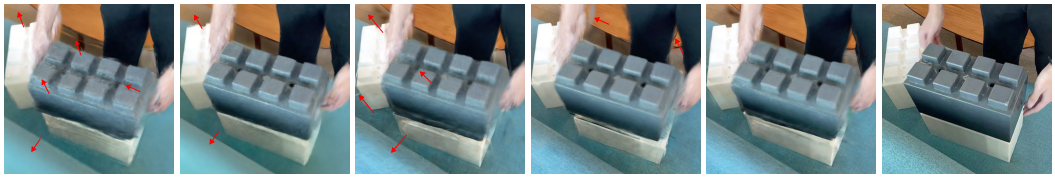


Figure 6: Qualitative evaluation of our ablation study with settings corresponding to Tab. 5.

Table 6: Quantitative results of an ablation study of the number of input frames used to personalise the diffusion model.

Method	PSNR-m	SSIM-m	LPIPS-m	PSNR-D	SSIM-D	LPIPS-D
3 frames	18.40	0.6577	0.3731	15.40	0.8832	0.4165
6 frames	18.51	0.6590	0.3714	15.59	0.8849	0.4132
Full sequence	19.00	0.6672	0.3623	16.32	0.8941	0.3921

a lower LPIPS score than our method, but the improved perceptual quality comes at the cost of reduced spatio-temporal consistency, which is reflected in the PSNR and SSIM scores, and also clearly shown in our supplementary video. In Table 3 we present quantitative results of ViDAR on the NVIDIA dataset [60]. Even though our method was designed to tackle highly ill-posed settings, it outperforms other methods on this simpler, forward-facing dataset. Similarly to DyCheck, we observe improvements both in full image and dynamic regions evaluations. We present a qualitative evaluation in Figure 4 (DyCheck) and Figure 5, where ViDAR demonstrates consistently superior visual quality and geometric consistency when compared to the best existing approaches. Although 2D image comparisons are indicative of performance, we encourage viewing our supplementary video results to appreciate the improvement in spatio-temporal consistency and visual quality over baselines. Moreover, we run ViDAR on the in-the-wild videos. We refer the reader to the Supplementary Material, where we present qualitative examples and no-reference metrics (VBench [8]) regarding this experiment. Similarly, the Supplementary Material shows the no-reference metrics evaluated on DyCheck dataset, indicating high temporal consistency of ViDAR outputs when compared to the state of the art.

Ablations - Contributions We quantitatively evaluate each of our contributions in an ablation study presented in Table 5. The penultimate row, w/o diffusion isolates the contribution of Tracking Based Gaussian Classification. The row w/o SO + DR + TGS shows a naive approach of using the diffused novel views directly as supervision for our monocular baseline without diffusion-aware reconstruction. Due to the spatio-temporal inconsistencies of the diffused outputs, this leads to a poor quality reconstruction, as shown in Figure 6. We show that removing dynamic reconstruction leads to blurry reconstruction in static regions, while removing sampled camera optimization leads to geometric inconsistencies. We also show that using our tracking based Gaussian classification reduces floaters. This ablation emphasises that all of the proposed contributions are important in achieving the final rendering quality.

Ablations - Input Video We also evaluate ViDAR in a scarce input setting. Instead of using a full input video sequence, we select 3 or 6 equally spaced frames in time (mimicking sparse static approaches) to train the personalised diffusion model. We use such enhancement on the full video sequence keeping all the other experiment settings unchanged from our full pipeline. We present the quantitative analysis in Table 6. We observe an increase in performance across all metrics with the increased number of frames saturating when using the whole video. Notably, both 3 and 6 frame setting provide an improvement upon baseline MoSca. With a lower number of frames, fine-tuning the personalised diffusion model can be performed quicker, thus providing a tradeoff between training speed and rendering quality.

5 Conclusion

We present ViDAR, a novel method for 4D reconstruction of scenes from monocular inputs. ViDAR leverages image diffusion models by conditioning on scene-specific features to recover fine-grained appearance details of novel viewpoints. ViDAR overcomes the spatio-temporal inconsistency of diffusion-based supervision via a diffusion-aware loss function and a camera pose optimisation strategy. We show that ViDAR outperforms all state-of-the-art baselines on the challening DyCheck dataset as well as the NVIDIA dataset, and we present a new benchmark to evaluate performance in dynamic regions.

Limitations: ViDAR limits the scope of diffusion to enhancing rendered images, which are limited by the initial accuracy of the 4D reconstruction, thus, cannot repair major geometrical artefacts.

References

[1] M.-Q. V. Bui, J. Park, J. Oh, and M. Kim. DyBluRF: Dynamic Deblurring Neural Radiance Fields for Blurry Monocular Video. *arXiv preprint arXiv:2312.13528*, 2023.

[2] A. Cao and J. CV. HexPlane: A Fast Representation for Dynamic Scenes. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2023.

[3] S. Fridovich-Keil, G. Meanti, F. R. Warburg, B. Recht, and A. Kanazawa. K-Planes: Explicit Radiance Fields in Space, Time, and Appearance. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2023.

[4] C. Gao, A. Saraf, J. Kopf, and J. Huang. Dynamic View Synthesis from Dynamic Monocular Video. In *International Conference on Computer Vision (ICCV)*, 2021.

[5] H. Gao, R. Li, S. Tulsiani, B. Russell, and A. Kanazawa. Monocular Dynamic View Synthesis: A Reality Check. In *Conference on Neural Information Processing Systems*, 2022.

[6] V. Guizilini, M. Z. Irshad, D. Chen, G. Shakhnarovich, and R. Ambrus. Zero-Shot Novel View and Depth Synthesis with Multi-View Geometric Diffusion. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

[7] Y.-H. Huang, Y.-T. Sun, Z. Yang, X. Lyu, Y.-P. Cao, and X. Qi. SC-GS: Sparse-Controlled Gaussian Splatting for Editable Dynamic Scenes. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2023.

[8] Z. Huang, Y. He, J. Yu, F. Zhang, C. Si, Y. Jiang, Y. Zhang, T. Wu, Q. Jin, N. Chanpaisit, Y. Wang, X. Chen, L. Wang, D. Lin, Y. Qiao, and Z. Liu. VBench: Comprehensive benchmark suite for video generative models. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2024.

[9] M. Kappel, F. Hahlbohm, T. Scholz, S. Castillo, C. Theobalt, M. Eisemann, V. Golyanik, and M. Magnor. D-NPC: Dynamic neural point clouds for non-rigid view synthesis from monocular video. *Proceedings of the Eurographics Conference (EG)*, 44, 2025.

[10] T. Karras, M. Aittala, T. Aila, and S. Laine. Elucidating the design space of diffusion-based generative models. In *Conference on Neural Information Processing Systems*, 2022.

[11] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics*, 42(4), July 2023.

[12] J. Lee, C. Won, H. Jung, I. Bae, and H.-G. Jeon. Fully Explicit Dynamic Guassian Splatting. In *Proceedings of the Neural Information Processing Systems*, 2024.

[13] J. Lei, Y. Weng, A. Harley, L. Guibas, and K. Daniilidis. MoSca: Dynamic gaussian fusion from casual videos via 4D motion scaffolds. *Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

[14] H. Li, H. Shi, W. Zhang, W. Wu, Y. Liao, L. Wang, L.-h. Lee, and P. Y. Zhou. Dreamscene: 3d gaussian-based text-to-3d scene generation via formation pattern sampling. In *European Conference on Computer Vision (ECCV)*, 2024.

[15] L. Li, Z. Shen, Z. Wang, L. Shen, and P. Tan. Streaming Radiance Fields for 3D Video Synthesis. In *Conference on Neural Information Processing Systems*, 2022.

[16] Z. Li, S. Niklaus, N. Snavely, and O. Wang. Neural Scene Flow Fields for Space-Time View Synthesis of Dynamic Scenes. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2021.

[17] Z. Li, Q. Wang, F. Cole, R. Tucker, and N. Snavely. DynIBaR: Neural Dynamic Image-Based Rendering. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2023.

[18] Z. Li, Z. Chen, Z. Li, and Y. Xu. Spacetime Gaussian Feature Splatting for Real-Time Dynamic View Synthesis. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2024.

[19] H. Liang, J. Ren, A. Mirzaei, A. Torralba, Z. Liu, I. Gilitschenski, S. Fidler, C. Oztireli, H. Ling, Z. Gojcic, and J. Huang. Feed-Forward Bullet-Time Reconstruction of Dynamic Scenes from Monocular Videos. *arXiv preprint arXiv:2412.03526*, 2024.

[20] Y. Liang, X. Yang, J. Lin, H. Li, X. Xu, and Y. Chen. Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2024.

[21] C. Lin, P. Pan, B. Yang, Z. Li, and Y. Mu. DiffSplat: Repurposing Image Diffusion Models for Scalable 3D Gaussian Splat Generation. In *International Conference on Learning Representations (ICLR)*, 2025.

[22] Y. Lin, Z. Dai, S. Zhu, and Y. Yao. Gaussian-Flow: 4D Reconstruction with Dynamic 3D Gaussian Particle. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2024.

[23] M. Liu, R. Shi, L. Chen, Z. Zhang, C. Xu, X. Wei, H. Chen, C. Zeng, J. Gu, and H. Su. One-2-3-45++: Fast Single Image to 3D Objects with Consistent Multi-View Generation and 3D Diffusion. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2024.

[24] M. Liu, C. Zeng, X. Wei, R. Shi, L. Chen, C. Xu, M. Zhang, Z. Wang, X. Zhang, I. Liu, H. Wu, and H. Su. MeshFormer: High-Quality Mesh Generation with 3D-Guided Reconstruction Model. In *Conference on Neural Information Processing Systems*, 2024.

[25] Q. Liu, Y. Liu, J. Wang, X. Lyu, P. Wang, W. Wang, and J. Hou. MoDGS: Dynamic Gaussian Splatting from Casually-captured Monocular Videos with Depth Priors. In *International Conference on Learning Representations (ICLR)*, 2025.

[26] Y.-L. Liu, C. Gao, A. Meuleman, H.-Y. Tseng, A. Saraf, C. Kim, Y.-Y. Chuang, J. Kopf, and J.-B. Huang. Robust dynamic radiance fields. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2023.

[27] J. Luiten, G. Kopanas, B. Leibe, and D. Ramanan. Dynamic 3D Gaussians: Tracking by Persistent Dynamic View Synthesis. In *International Conference on 3D Vision (3DV)*, 2024.

[28] X. Miao, Y. Bai, H. Duan, F. Wan, Y. Huang, Y. Long, and Y. Zheng. CTNeRF: Cross-time Transformer for dynamic neural radiance field from monocular video. *Pattern Recognition*, 156:110729, 2024.

[29] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision (ECCV)*, 2020.

[30] J. Park, M.-Q. V. Bui, J. L. G. Bello, J. Moon, J. Oh, and M. Kim. SplineGS: Robust Motion-Adaptive Spline for Real-Time Dynamic 3D Gaussians from Monocular Video. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

[31] K. Park, U. Sinha, J. T. Barron, S. Bouaziz, D. B. Goldman, S. M. Seitz, and R. Martin-Brualla. Nerfies: Deformable Neural Radiance Fields. *International Conference on Computer Vision (ICCV)*, 2021.

[32] K. Park, U. Sinha, P. Hedman, J. T. Barron, S. Bouaziz, D. B. Goldman, R. Martin-Brualla, and S. M. Seitz. HyperNeRF: a higher-dimensional representation for topologically varying neural radiance fields. *ACM Transactions on Graphics*, 40(6), 2021.

[33] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. In *International Conference on Learning Representations (ICLR)*, 2024.

[34] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall. DreamFusion: Text-to-3D using 2D Diffusion. In *International Conference on Learning Representations (ICLR)*, 2023.

[35] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer. D-NeRF: Neural Radiance Fields for Dynamic Scenes. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2021.

[36] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2021.

[37] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman. DreamBooth: Fine Tuning Text-to-image Diffusion Models for Subject-Driven Generation. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2023.

[38] R. Shao, Z. Zheng, H. Tu, B. Liu, H. Zhang, and Y. Liu. Tensor4D: Efficient Neural 4D Decomposition for High-fidelity Dynamic Reconstruction and Rendering. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2023.

[39] R. Shaw, M. Nazarczuk, J. Song, A. Moreau, S. Catley-Chandar, H. Dhamo, and E. Pérez-Pellitero. SWinGS: Sliding Windows for Dynamic 3D Gaussian Splatting. In *European Conference on Computer Vision (ECCV)*, 2024.

[40] J. Shriram, A. Trevithick, L. Liu, and R. Ramamoorthi. RealmDreamer: Text-Driven 3D Scene Generation with Inpainting and Depth Diffusion. In *International Conference on 3D Vision (3DV)*, 2025.

[41] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[42] C. Stearns, A. W. Harley, M. Uy, F. Dubost, F. Tombari, G. Wetzstein, and L. Guibas. Dynamic Gaussian Marbles for Novel View Synthesis of Casual Monocular Videos. In *SIGGRAPH Asia*, 2024.

[43] J. Tang, Z. Chen, X. Chen, T. Wang, G. Zeng, and Z. Liu. LGM: Large Multi-View Gaussian Model for High-Resolution 3D Content Creation. In *European Conference on Computer Vision (ECCV)*, 2024.

[44] C. Wang, P. Zhuang, A. Siarohin, J. Cao, G. Qian, H.-Y. Lee, and S. Tulyakov. Diffusion Priors for Dynamic View Synthesis from Monocular Videos. *arXiv preprint arXiv:2401.05583*, 2024.

[45] F. Wang, S. Tan, X. Li, Z. Tian, and H. Liu. Mixed Neural Voxels for Fast Multi-view Video Synthesis. In *International Conference on Computer Vision (ICCV)*, 2023.

[46] Q. Wang, V. Ye, H. Gao, W. Zeng, J. Austin, Z. Li, and A. Kanazawa. Shape of Motion: 4D Reconstruction from a Single Video. In *arXiv preprint arXiv:2407.13764*, 2024.

[47] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 13(4), 2004.

[48] T. Wimmer, M. Oechsle, M. Niemeyer, and F. Tombari. Gaussians-to-Life: Text-Driven Animation of 3D Gaussian Splatting Scenes. In *International Conference on 3D Vision (3DV)*, 2025.

[49] G. Wu, T. Yi, J. Fang, L. Xie, X. Zhang, W. Wei, W. Liu, Q. Tian, and X. Wang. 4D Gaussian Splatting for Real-Time Dynamic Scene Rendering. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2024.

[50] J. Z. Wu, Y. Zhang, H. Turki, X. Ren, J. Gao, M. Z. Shou, S. Fidler, Z. Gojcic, and H. Ling. Difix3d+: Improving 3d reconstructions with single-step diffusion models. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

[51] R. Wu, R. Gao, B. Poole, A. Trevithick, C. Zheng, J. T. Barron, and A. Holynski. CAT4D: Create Anything in 4D with Multi-View Video Diffusion Models. *arXiv:2411.18613*, 2024.

[52] R. Wu, B. Mildenhall, P. Henzler, K. Park, R. Gao, D. Watson, P. P. Srinivasan, D. Verbin, J. T. Barron, B. Poole, and A. Holynski. ReconFusion: 3D Reconstruction with Diffusion Priors. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2024.

[53] J. Xu, W. Cheng, Y. Gao, X. Wang, S. Gao, and Y. Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024.

[54] Y. Xu, H. Tan, F. Luan, S. Bi, P. Wang, J. Li, Z. Shi, K. Sunkavalli, G. Wetzstein, Z. Xu, and K. Zhang. DMV3D: Denoising Multi-View Diffusion using 3D Large Reconstruction Model. In *International Conference on Learning Representations (ICLR)*, 2024.

[55] C. Yang, S. Li, J. Fang, R. Liang, L. Xie, X. Zhang, W. Shen, and Q. Tian. GaussianObject: High-Quality 3D Object Reconstruction from Four Views with Gaussian Splatting. In *SIGGRAPH Asia*, 2024.

[56] J. Yang, M. Gao, Z. Li, S. Gao, F. Wang, and F. Zheng. Track Anything: Segment Anything Meets Videos. *arXiv preprint arXiv:2304.11968*, 2023.

[57] S. Yang, W. Yu, J. Zeng, J. Lv, K. Ren, C. Lu, D. Lin, and J. Pang. Novel demonstration generation with gaussian splatting enables robust one-shot manipulation. *arXiv preprint arXiv:2504.13175*, 2025.

[58] Z. Yang, X. Gao, W. Zhou, S. Jiao, Y. Zhang, and X. Jin. Deformable 3D Gaussians for High-Fidelity Monocular Dynamic Scene Reconstruction. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2024.

[59] T. Yi, J. Fang, J. Wang, G. Wu, L. Xie, X. Zhang, W. Liu, Q. Tian, and X. Wang. GaussianDreamer: Fast Generation from Text to 3D Gaussians by Bridging 2D and 3D Diffusion Models. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2024.

[60] J. S. Yoon, K. Kim, O. Gallo, H. S. Park, and J. Kautz. Novel view synthesis of dynamic scenes with globally coherent depths from a monocular camera. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2020.

[61] H. Yu, C. Wang, P. Zhuang, W. Menapace, A. Siarohin, J. Cao, L. A. Jeni, S. Tulyakov, and H.-Y. Lee. 4Real: Towards Photorealistic 4D Scene Generation via Video Diffusion Models. In *Conference on Neural Information Processing Systems*, 2024.

[62] Y. Zeng, Y. Jiang, S. Zhu, Y. Lu, Y. Lin, H. Zhu, W. Hu, X. Cao, and Y. Yao. STAG4D: Spatial-Temporal Anchored Generative 4D Gaussians. In *European Conference on Computer Vision (ECCV)*, 2024.

[63] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2018.

[64] X. Zhao, A. Colburn, F. Ma, M. Ángel Bautista, J. M. Susskind, and A. G. Schwing. Pseudo-Generalized Dynamic View Synthesis from a Video. In *International Conference on Learning Representations (ICLR)*, 2024.

[65] K. Zhou, J.-X. Zhong, S. Shin, K. Lu, Y. Yang, A. Markham, and N. Trigoni. DynPoint: dynamic neural point for view synthesis. In *Conference on Neural Information Processing Systems*, 2023.

[66] R. Zhu, Y. Liang, H. Chang, J. Deng, J. Lu, W. Yang, T. Zhang, and Y. Zhang. MotionGS: Exploring Explicit Motion Guidance for Deformable 3D Gaussian Splatting. In *Conference on Neural Information Processing Systems*, 2024.

[67] Z.-X. Zou, Z. Yu, Y.-C. Guo, Y. Li, D. Liang, Y.-P. Cao, and S.-H. Zhang. Triplane Meets Gaussian Splatting: Fast and Generalizable Single-View 3D Reconstruction with Transformers. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2024.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The claims made in the abstract and the introduction (explicitly stated) reflect the contributions and are validated through extensive experimentation (Tables 1, 2, 5, Figures 4, 6).

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Limitations are discussed in the main manuscript and supplementary material.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [NA]

Justification: The paper does not include theoretical results

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes] ,

   Justification: Based on the method section of the main manuscript it would be possible re-implement the approach described in the paper and achieve similar results. Regardless, further implementation details are provided in the supplementary.

   Guidelines:

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

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

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

Answer: [No]

Justification: We plan to release code upon acceptance and pending internal approval procedures.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: Core details are provided in the main manuscript and all other details are provided in the supplementary.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [No]

Justification: Our paper builds upon MoSca [13] and DyCheck[5], among many other works, and follows their experimental setup. In our approach we run the experiments with a fixed seed (same as MoSca), which we report in the supplementary for reproducibility.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. **Experiments compute resources**

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

   Answer: [Yes] ,

   Justification: Computational resources, memory and execution time are discussed in the supplementary.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The authors reviewed the NeurIPS Code of Ethics and confirm that the conducted research adheres to the code.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Broader impacts are discussed in the supplementary material.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

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

11. **Safeguards**

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

    Answer: [NA]

    Justification: The paper does not pose a high risk for misuse.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: All code, data and models used are properly credited and any license terms are properly respected.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

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

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

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

    Answer: [NA]

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

    Guidelines:

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

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

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

    Answer: [NA]

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

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [NA]

    Justification: This research does not involve LLMs.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

ViDAR: Video Diffusion-Aware 4D Reconstruction From Monocular Inputs

Supplementary Material

A Additional Results

In this section, we include additional qualitative and quantitative evaluation of ViDAR.

A.1 Further Qualitative Evaluation

We present additional qualitative evaluation of ViDAR compared to MoSca and Shape Of Motion on the qualitative example scenes from the DyCheck dataset in Fig. 7. Our results show consistently greater geometric consistency and visual quality compared to the other approaches.

A.2 Per-Scene Results

We provide a detailed quantitative evaluation for every scene of the DyCheck dataset in Tables 7 and 8, in half and full resolution respectively. As in Tables 1 and 2, we compute PSNR, SSIM and LPIPS on the co-visibility masked regions of the test views, which we denote with an *-m* addendum to each metric, as well as on the dynamic masked regions of the test views which we denote with a *-D*. With a few exceptions (e.g. Apple, co-visibility), ViDAR is consistently the best performing method. Similarly, we show detailed results of the NVIDIA dataset in Table 9.

A.3 In-the-wild videos

Finally, we evaluated ViDAR on the set of in-the-wild videos proposed by MoSca. Table 10 presents a selection of non-reference video quality metrics comparing the input sequence against MoSca and ViDAR. The metrics were proposed by a comprehensive video quality assessment suite - VBench and consist of *Subject Consistency, Background Consistency, Motion Smoothness, Aesthetic Quality, Imaging Quality*. Further, the qualitative examples of the scenes are shown in Figure 8.

A.4 Ablation - generalisable enhancement

In addition, we test our pipeline with the use of a generalisable diffusion model for enhancement. To this end, we substitute the personalised diffusion model with a generalisable one proposed in Difix3D+. Further, we enhance the mutli-view supervision images as in the original experiment and keep the rest of the pipeline intact. We use two available settings of Difix3D+, no-reference enhancement, and a reference one, where we provide a frame from the input video as the enhancement guidance. We show the results of this ablation in Table 11. This indicates that the use of personalised diffusion provides samples more suitable for the reconstruction in the proposed pipeline.

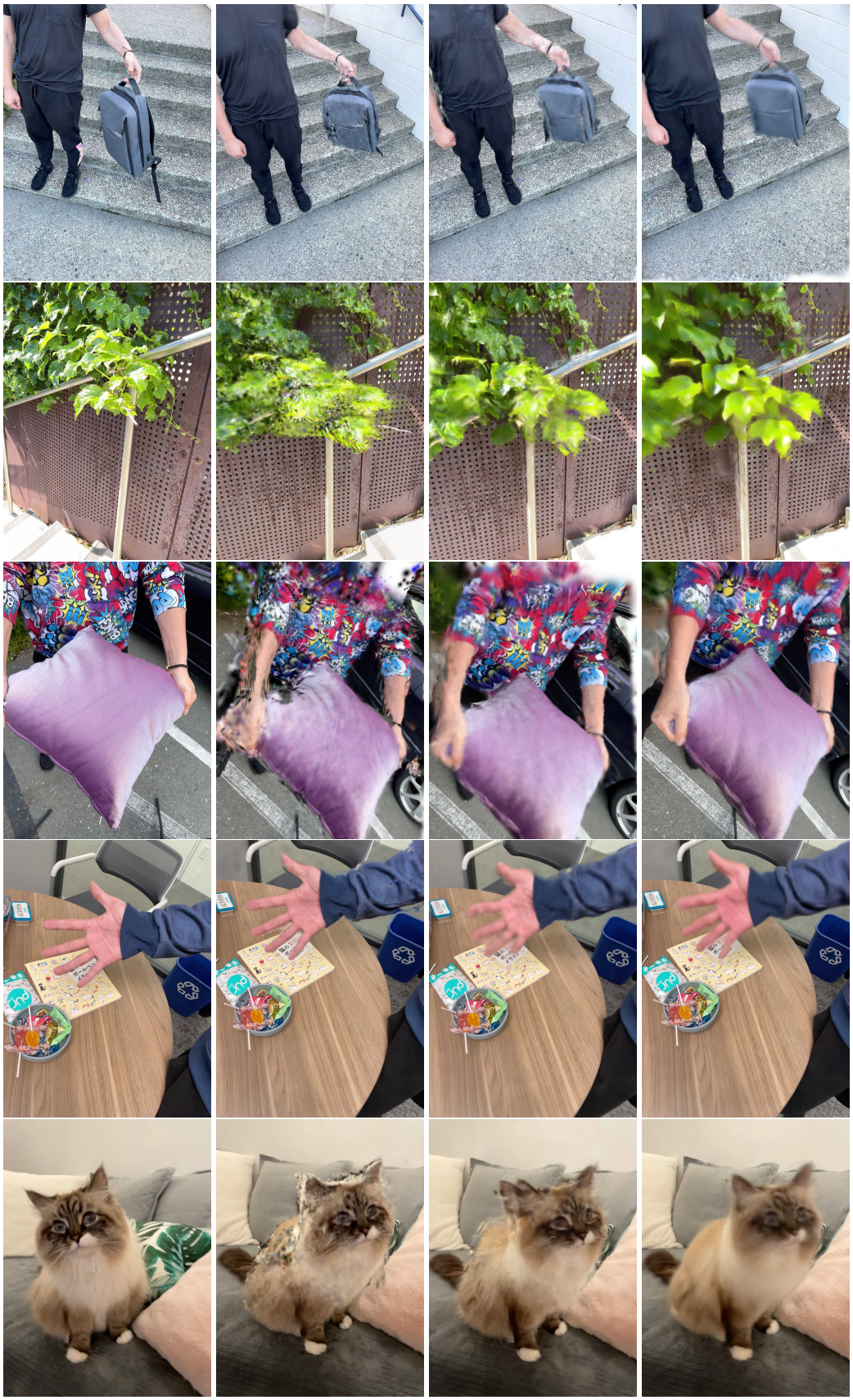
A.5 DyCheck - video consistency metrics

In Table 12 we provide no-reference video consistency metrics on the DyCheck dataset. The results indicate that while raw diffusion output is characterised by low temporal consistency, ViDAR is capable of utilising strong support for ambiguous geometry from diffusion outputs in order to produce high-quality and temporally consistent results.

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

Table 7: Per-scene quantitative evaluation of ViDAR against state-of-the-art methods on the DyCheck dataset at half resolution. Best, second and third results are highlighted in red, orange and yellow respectively.

	Method	PSNR-m↑	SSIM-m↑	LPIPS-m↓	PSNR-D↑	SSIM-D↑	LPIPS-D↓
Apple	T-NeRF [5]	17.43	0.7285	0.5081	13.63	0.9433	0.3108
	Nerfies [31]	17.54	0.7505	0.4785	13.63	0.9437	0.3321
	HyperNeRF [32]	17.64	0.7433	0.4775	13.36	0.9417	0.3362
	Gaussian Marbles [42]	17.90	0.7328	0.4716	14.56	0.9370	0.3229
	SoM [46]	18.95	0.8111	0.2917	14.88	0.9419	0.3016
	MoSca [13]	19.40	0.8074	0.3392	16.97	0.9580	0.2176
	Ours	19.18	0.8008	0.3149	17.75	0.9616	0.1893
Block	T-NeRF [5]	17.52	0.6688	0.3460	14.07	0.8591	0.3474
	Nerfies [31]	16.61	0.6393	0.3893	13.31	0.8433	0.4068
	HyperNeRF [32]	17.54	0.6702	0.3312	13.73	0.8525	0.3483
	Gaussian Marbles [42]	16.95	0.6509	0.3788	13.81	0.8480	0.3750
	SoM [46]	17.99	0.6634	0.2608	15.23	0.8596	0.3378
	MoSca [13]	18.11	0.6801	0.3416	15.32	0.8699	0.3499
	Ours	18.91	0.6901	0.2168	15.93	0.8864	0.3052
Paper	T-NeRF [5]	17.55	0.3672	0.2577	12.50	0.9730	0.3149
	Nerfies [31]	17.34	0.3783	0.2111	10.84	0.9710	0.3929
	HyperNeRF [32]	17.38	0.3819	0.2086	10.64	0.9706	0.4040
	Gaussian Marbles [42]	16.62	0.3219	0.3517	11.68	0.9710	0.2680
	SoM [46]	20.85	0.6725	0.1536	12.90	0.9738	0.2634
	MoSca [13]	22.24	0.7450	0.1617	14.32	0.9789	0.2832
	Ours	22.48	0.7516	0.1287	15.58	0.9807	0.3080
Space-out	T-NeRF [5]	17.71	0.5914	0.3768	14.52	0.8620	0.3649
	Nerfies [31]	17.79	0.6217	0.3032	13.85	0.8578	0.3407
	HyperNeRF [32]	17.93	0.6054	0.3203	14.26	0.8597	0.3379
	Gaussian Marbles [42]	15.32	0.5512	0.4235	11.15	0.8488	0.4278
	SoM [46]	19.64	0.6313	0.2374	15.72	0.8805	0.2429
	MoSca [13]	20.35	0.6582	0.2702	17.57	0.8985	0.2073
	Ours	21.58	0.6890	0.2010	18.77	0.9091	0.2306
Spin	T-NeRF [5]	19.16	0.5672	0.4427	15.65	0.9090	0.3394
	Nerfies [31]	18.38	0.5846	0.3087	14.11	0.8899	0.3868
	HyperNeRF [32]	19.20	0.5614	0.3254	15.65	0.9060	0.3230
	Gaussian Marbles [42]	18.31	0.5461	0.3558	15.68	0.8963	0.3950
	SoM [46]	21.05	0.7798	0.1698	18.44	0.9180	0.2815
	MoSca [13]	21.06	0.7100	0.2084	18.85	0.9240	0.2688
	Ours	21.28	0.7209	0.1853	19.37	0.9327	0.2566
Teddy	T-NeRF [5]	13.71	0.5695	0.4286	13.49	0.6198	0.3730
	Nerfies [31]	13.65	0.5572	0.3716	13.33	0.6050	0.3280
	HyperNeRF [32]	13.97	0.5678	0.3498	13.65	0.6168	0.3028
	Gaussian Marbles [42]	13.65	0.5432	0.4369	13.29	0.5972	0.3961
	SoM [46]	14.00	0.5462	0.3399	13.68	0.5998	0.3335
	MoSca [13]	15.09	0.6133	0.3587	14.84	0.6679	0.3146
	Ours	15.97	0.6416	0.3096	15.62	0.6844	0.2901
Wheel	T-NeRF [5]	15.65	0.5481	0.2925	13.19	0.8162	0.3937
	Nerfies [31]	13.82	0.4580	0.3097	11.15	0.7870	0.4805
	HyperNeRF [32]	13.99	0.4550	0.3102	11.60	0.7913	0.4385
	Gaussian Marbles [42]	16.02	0.5416	0.3398	9.99	0.8175	0.3926
	SoM [46]	17.86	0.6695	0.2144	12.75	0.8338	0.3446
	MoSca [13]	17.95	0.6852	0.2309	11.57	0.8310	0.3918
	Ours	18.46	0.6943	0.2058	12.23	0.8402	0.3754



Input | Shape Of Motion [46] | MoSca [13] | Ours

Figure 7: Qualitative evaluation of our method against benchmark methods on the DyCheck qualitative example set.

Table 8: Per-scene quantitative evaluation of ViDAR against state-of-the-art methods on the DyCheck dataset at full resolution. Best, second and third results are highlighted in red, orange and yellow respectively.

	Method	PSNR-m↑	SSIM-m↑	LPIPS-m↓	PSNR-D↑	SSIM-D↑	LPIPS-D↓
Apple	Gaussian Marbles [42]	16.84	0.7022	0.6849	14.65	0.9495	0.4281
	SoM [46]	17.74	0.7535	0.4946	14.88	0.9540	0.4036
	MoSca [13]	18.19	0.7486	0.5651	17.21	0.9681	0.3073
	Ours	18.02	0.7466	0.5359	18.07	0.9694	0.2559
Block	Gaussian Marbles [42]	16.50	0.6492	0.5065	13.49	0.8660	0.4987
	SoM [46]	17.42	0.6566	0.3879	14.60	0.8759	0.4667
	MoSca [13]	17.56	0.6710	0.4658	14.97	0.8848	0.4808
	Ours	18.43	0.6722	0.3932	15.54	0.8898	0.4100
Paper	Gaussian Marbles [42]	15.96	0.2959	0.5778	11.30	0.9722	0.5094
	SoM [46]	19.65	0.5518	0.2035	13.03	0.9737	0.4861
	MoSca [13]	20.82	0.6289	0.2412	14.04	0.9770	0.5498
	Ours	21.06	0.6477	0.1923	14.99	0.9777	0.5003
Space-out	Gaussian Marbles [42]	15.19	0.5603	0.5434	11.07	0.8671	0.5334
	SoM [46]	19.54	0.6178	0.3667	16.11	0.8939	0.3363
	MoSca [13]	19.93	0.6280	0.4060	17.17	0.8988	0.3207
	Ours	21.15	0.6443	0.3278	18.56	0.9079	0.3356
Spin	Gaussian Marbles [42]	17.84	0.5154	0.4784	15.79	0.9143	0.4420
	SoM [46]	20.57	0.7323	0.2663	18.65	0.9339	0.3535
	MoSca [13]	20.61	0.6769	0.3108	18.50	0.9371	0.3657
	Ours	20.69	0.6851	0.2868	19.37	0.9445	0.3094
Teddy	Gaussian Marbles [42]	13.01	0.5496	0.6559	13.05	0.6269	0.5815
	SoM [46]	13.58	0.5518	0.5377	13.40	0.6251	0.4928
	MoSca [13]	14.52	0.5925	0.5777	14.41	0.6692	0.4986
	Ours	15.63	0.6238	0.4786	15.46	0.6892	0.4133
Wheel	Gaussian Marbles [42]	15.55	0.5310	0.5295	9.89	0.8291	0.5473
	SoM [46]	17.38	0.6318	0.3456	13.05	0.8393	0.5037
	MoSca [13]	17.49	0.6460	0.3684	11.42	0.8395	0.5665
	Ours	18.00	0.6505	0.3215	12.21	0.8469	0.5203

B Implementation Details

In this section, we provide any implementation details not included in the main manuscript.

B.1 Monocular Reconstruction

We implement the monocular reconstruction step directly as MoSca [13], keeping the original hyperparameters intact. We substitute dynamic masks estimated from epipolar error by masks obtained from Track Anything [56].

B.2 Personalised Diffusion Model

We train our personalised diffusion model with a Dreambooth [37] approach implemented in the `diffusers`[2] library as a LoRA fine-tuning process. We use the default implementation of the SDXL model with default parameters. We change the resolution to match our input resolution (720x960). Similarly, we change the number of training iterations from the default 500 to 5000, in response to the default model being suitable for personalisation with a smaller number of images (5-40), as opposed to our inputs (ranging above 400).

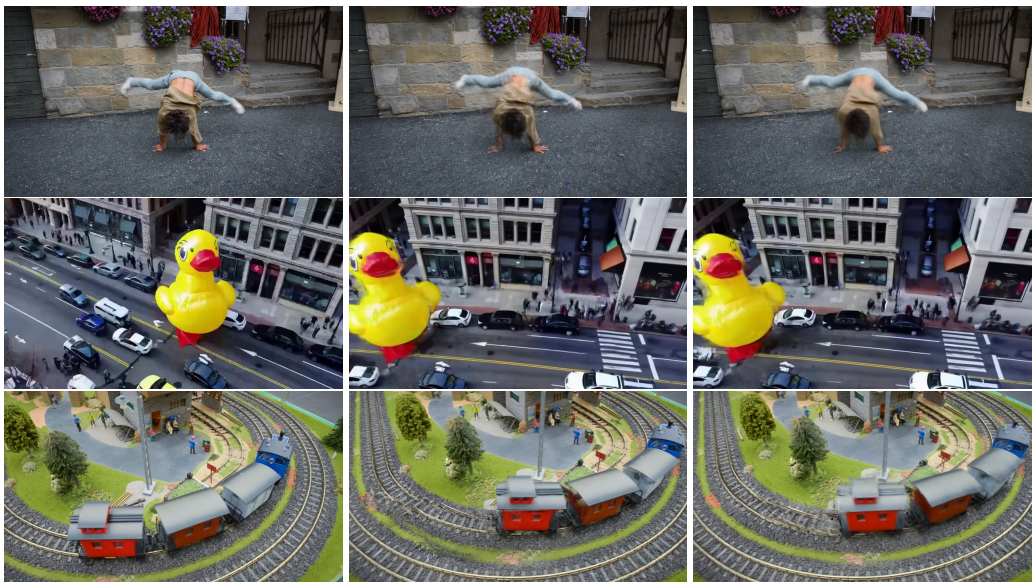
[2] `https://huggingface.co/docs/diffusers`

Table 9: Per-scene quantitative evaluation of ViDAR against state-of-the-art methods on the NVIDIA dataset at full resolution. Best, second and third results are highlighted in red, orange and yellow respectively.

	Method	PSNR-m↑	SSIM-m↑	LPIPS-m↓	PSNR-D↑	SSIM-D↑	LPIPS-D↓
Balloon1	DynNeRF [4]	22.36	0.7748	0.1015	18.68	0.8805	0.1950
	RoDynRF [26]	22.37	0.7818	0.0998	18.95	0.8800	0.1876
	NSFF [16]	21.96	0.7013	0.2119	18.66	0.8858	0.2573
	MoSca [13]	23.58	0.8008	0.1002	19.80	0.8811	0.2149
	Ours	23.98	0.8173	0.0817	20.07	0.8953	0.1686
Balloon2	DynNeRF [4]	27.06	0.8591	0.0496	20.73	0.9640	0.0543
	RoDynRF [26]	26.19	0.8459	0.0532	19.87	0.9549	0.0654
	NSFF [16]	24.27	0.7314	0.2144	19.88	0.9528	0.1448
	MoSca [13]	27.80	0.8755	0.0538	21.98	0.9696	0.0633
	Ours	28.30	0.8857	0.0490	22.14	0.9722	0.0519
Jumping	DynNeRF [4]	24.68	0.8417	0.0863	18.35	0.9372	0.1603
	RoDynRF [26]	25.66	0.8532	0.0691	19.49	0.9450	0.1322
	NSFF [16]	24.65	0.8125	0.1465	18.27	0.9359	0.2090
	MoSca [13]	25.02	0.8135	0.0914	18.74	0.9368	0.1452
	Ours	25.74	0.8355	0.0715	19.02	0.9449	0.1422
Playground	DynNeRF [4]	24.15	0.8492	0.0767	17.10	0.9775	0.1110
	RoDynRF [26]	24.96	0.8993	0.0464	16.28	0.9732	0.1136
	NSFF [16]	21.22	0.7047	0.2168	16.76	0.9790	0.1629
	MoSca [13]	24.25	0.8872	0.0500	17.13	0.9773	0.1036
	Ours	24.79	0.8939	0.0479	17.28	0.9776	0.1023
Skating	DynNeRF [4]	32.66	0.9514	0.0328	21.05	0.9908	0.1202
	RoDynRF [26]	28.68	0.9394	0.0387	14.88	0.9842	0.2283
	NSFF [16]	29.29	0.8890	0.1208	20.03	0.9908	0.1768
	MoSca [13]	33.41	0.9502	0.0305	22.35	0.9927	0.0988
	Ours	34.07	0.9554	0.0248	22.34	0.9929	0.0959
Truck	DynNeRF [4]	28.56	0.8722	0.0811	26.32	0.9702	0.0832
	RoDynRF [26]	29.13	0.8996	0.0616	27.85	0.9779	0.0681
	NSFF [16]	25.96	0.7750	0.1724	26.38	0.9699	0.1362
	MoSca [13]	27.77	0.8608	0.0792	27.77	0.8608	0.0802
	Ours	28.34	0.8760	0.0724	28.08	0.9793	0.0544
Umbrella	DynNeRF [4]	23.26	0.7107	0.1303	17.08	0.9548	0.1638
	RoDynRF [26]	24.26	0.7572	0.0850	19.78	0.9653	0.1006
	NSFF [16]	22.97	0.6439	0.2743	16.57	0.9510	0.2625
	MoSca [13]	25.18	0.7669	0.0907	20.36	0.9656	0.1609
	Ours	25.19	0.7630	0.0817	20.51	0.9672	0.1260

Table 10: No-reference evaluation (VBench) of ViDAR against MoSca and the input video. The provided metrics are part of VBench suite.

	Method	Subj. cons. ↑	Bg. cons. ↑	Motion ↑	Aesthetic ↑	Imaging ↑
breakdance	Input	0.9199	0.9577	0.9931	0.4507	0.7052
	MoSca	0.8795	0.9506	0.9924	0.4259	0.5565
	Ours	0.8963	0.9528	0.9925	0.4313	0.5843
duck	Input	0.9860	0.9447	0.9944	0.6272	0.7236
	MoSca	0.9561	0.9204	0.9937	0.5855	0.7045
	Ours	0.9653	0.9221	0.9939	0.5870	0.6966
shiba	Input	0.8907	0.9690	0.9976	0.4913	0.7470
	MoSca	0.8724	0.9596	0.9959	0.4363	0.6846
	Ours	0.8891	0.9657	0.9966	0.4842	0.6910
train	Input	0.9585	0.9790	0.9968	0.6858	0.7319
	MoSca	0.9464	0.9709	0.9953	0.6386	0.6730
	Ours	0.9521	0.9772	0.9955	0.6782	0.6835
Average	Input	0.9388	0.9626	0.9955	0.5637	0.7269
	MoSca	0.9136	0.9504	0.9943	0.5216	0.6547
	Ours	0.9257	0.9544	0.9947	0.5452	0.6638



Input MoSca [13] Ours

Figure 8: Qualitative evaluation of our method against MoSca on in-the-wild videos.

Table 11: Quantitative results of an ablation study with the use of generalisable enhancement model - Difix3D+.

Method	PSNR-m	SSIM-m	LPIPS-m	PSNR-D	SSIM-D	LPIPS-D
Difix3D+ - no ref	18.30	0.6558	0.3600	15.26	0.8837	0.3917
Difix3D+ - ref	18.27	0.6555	0.3625	15.25	0.8883	0.3923
Ours	19.00	0.6672	0.3623	16.32	0.8941	0.3921

Table 12: No-reference evaluation (VBench) of ViDAR against MoSca, CAT4D and raw diffusion output. The provided metrics are part of VBench suite.

Method	Subj. cons. ↑	Bg. cons. ↑	Motion ↑	Aesthetic ↑	Imaging ↑
Raw diffusion	0.9333	0.9273	0.9630	0.4020	0.5720
MoSca	0.9375	0.9323	0.9911	0.4197	0.5535
CAT4D	0.9440	0.9394	0.9939	0.3962	0.5521
Ours	0.9460	0.9439	0.9941	0.4262	0.5588
GT	0.9477	0.9489	0.9969	0.4577	0.7534

B.3 Camera Sampling

To obtain a set of varying samples for multi-view supervision, we propose a camera sampling strategy based on extreme poses within the input trajectory.

Given the set of input camera poses (position and orientation), we calculate a mean camera pose. Then, we establish a sphere approximating the surface established by the input trajectory, assuming that a target dynamic object is being tracked by the recording. Finally, we select two views in the input trajectory that, when projected on the sphere, are characterised by the largest longitudinal displacement. These constitute the extreme camera poses.

Thereafter, for each time step spanning the whole time range of the input video, we sample the following new cameras:

- Two random camera poses from the input trajectory are selected, and a new camera pose is calculated as their mean, and random noise is added. Total cameras: 4
- For each of the two extreme views, a random camera pose from the input trajectory is selected, and a new camera pose is calculated as their weighted average, and random noise is added, with the weight increasing towards the extreme views. Total cameras: 12
- The extreme camera views. Total cameras: 2

This constitutes our set of 18 new training cameras for each timestep of the input video $c_m \in \mathcal{C}_{sample}$.

B.4 Multi-View Sample Enhancement

Having sampled a set of new trajectories, we render them with the previously trained monocular reconstruction model, in such way we obtain a set of degraded images $\{R_{m,t}\}$. To perform the enhancement as described in Section 3.2, we utilise the Image2Image translation approach as implemented in `diffusers`.

B.5 Diffusion-Aware Reconstruction

We increase the total number of iterations from 8000 to 40000 in order to train on the additional generated data. During optimisation, we run two separate forward and backward passes, the first for sampled camera pose optimisation and the second for optimising the Gaussians and input camera poses. At each iteration, we randomly select two of the sampled camera poses which correspond to the same time step as the input camera. During the first pass, we render the images and compute the mean of the camera losses $\mathcal{L}_{cam}$ for both of the sampled cameras and update only the sampled camera poses $\mathcal{C}_{sample}$. During the second pass, we re-render the images using the updated camera poses and compute the dynamic loss $\mathcal{L}_{dyn}$ using the dynamic region masks. This loss is added to the existing monocular losses and is used to update the input camera poses $\mathcal{C}_{inp}$ and the Gaussians $\mathcal{G}$.

C Limitations

As mentioned in Section 5, the main limitation of our approach lies in the dependency on the underlying monocular reconstruction model as a starting point. If the monocular reconstruction model (in our work, MoSca) fails catastrophically, ViDAR will not be able to enhance the reconstruction,

e.g. if MoSca reconstructs a human with a detached arm, ViDAR cannot reattach the arm if it is too far away. This is due to the conditioning of the diffusion model on the rendering from initial reconstruction. Similarly, our work does not explore the extrapolation of the scene reconstruction through diffusion outpainting of the unseen areas.

D Broader Impacts

Our work focuses on improving the reconstruction of dynamic scenes in a monocular camera setting. We believe this limits the potential negative societal impacts, namely, ViDAR does not hallucinate in the regions outside the observed scene and requires a monocular prior on the geometry. This reduces the potential for misuse via the generation of fake content.

E Computational Resources

Our approach does not require a large amount of computational resources, as we use a single graphics card characterised by 60 TFLOPS at fp32. Our diffusion enhancement stage requires 2 hours to finetune and generate the multi-view images and we perform diffusion-aware reconstruction for a further 2 hours, bringing the total compute required to 4 GPU hours.