SCR4TCH: SELF-SUPERVISED 4D RECONSTRUCTION FROM RAW VIDEO ALONE

Anonymous authors
Paper under double-blind review

ABSTRACT

Self-supervised learning has enabled powerful visual representations at scale, yet its potential for understanding the 3D structure and dynamics of the world directly from raw video remains largely unrealized. We present *Scr4tch*, a model that learns geometrically grounded 4D representations entirely from *scratch* using *uncalibrated* monocular videos. Without requiring camera calibration, geometric annotations, or pretrained geometric models, *Scr4tch* unlocks self-supervised learning for 4D understanding from large-scale Internet videos. Our key idea is to learn 4D representations that encode cross-view relationships through latent reconstruction, while constraining how motion is modeled to separate camera motion from scene dynamics and suppress appearance shortcuts. Joint geometric optimization then grounds these representations in an explicit, deformable 4D scene. To train *Scr4tch*, we additionally collect 2,154 hours of challenging, casually captured videos from YouTube without offline geometric preprocessing. Experiments demonstrate accurate camera estimation, 4D reconstruction, and high-quality novel view synthesis on unseen videos, establishing a path toward scalable self-supervised learning of the dynamic world.

1 INTRODUCTION

Learning representations that capture the structure and dynamics of the world from raw videos is a fundamental challenge in self-supervised visual learning. Video offers a natural source of supervision: as cameras and objects move, successive images reveal how a scene persists and changes across space and time. Internet video supplies these observations at scale, but typically lacks camera calibration and geometric annotations.

Existing approaches only partially close this gap. Generalizable 4D reconstruction models learn from synthetic multi-view animations or calibrated videos (Ren et al., 2024; Xu et al., 2025), while methods that remove camera annotations often retain dependence on pretrained geometric models (Yuan et al., 2025; Balice et al., 2026). These dependencies shift part of the learning problem into data preparation or previously acquired geometric knowledge. Image-only self-supervised reconstruction offers an alternative, but recent advances primarily learn representations of static scenes (Jiang et al., 2025; Wang et al., 2025b). Even methods that learn from dynamic video may absorb or remove moving content to reconstruct a static background (Prestel et al., 2026; Chen et al., 2026). Learning scene structure and dynamics jointly from raw video alone remains an open challenge, due to the ambiguity between camera motion and scene motion.

A raw video does not label which image changes arise from viewpoint, scene motion, or visibility. In explicit reconstruction, scene deformation can reproduce camera-induced image changes even when the estimated camera poses remain unchanged. In latent reconstruction, motion features inferred from target images and passed to the renderer can additionally convey target appearance, allowing reconstruction to bypass camera estimation (Prestel et al., 2026). The capacity needed to represent scene motion varies across videos and is unknown in advance; excess capacity can therefore absorb viewpoint changes even in static regions, leaving camera estimates collapsed. Yet static structure offers a useful compression principle: given a shared scene representation, a small set of camera parameters can explain viewpoint changes across all static regions (Wang et al., 2025b). Restricting motion capacity thus encourages the model to explain these changes through cameras and reserve motion features for scene dynamics.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We present *Scr4tch*, a model that learns 4D representations entirely from scratch using only monocular RGB videos, without camera calibration, geometric annotations, or pretrained geometric models. Our key idea is to exploit this compression principle to separate camera motion from scene dynamics through latent reconstruction, and then ground the learned representations in explicit 4D geometry. We implement this idea through two-stage training. Latent pretraining compresses motion tokens using nested dropout following FlexTok (Bachmann et al., 2025), while explicit alignment jointly optimizes cameras and deformable 4D Gaussians through rendering and geometric regularization.

To train *Scr4tch*, we additionally collect 2,154 hours of challenging, casually captured videos from YouTube without offline geometric preprocessing. The resulting feed-forward model predicts cameras, motion masks, and deformable 4D Gaussian scenes from unseen videos and supports novel view synthesis (NVS). By removing external geometric prerequisites from learning, *Scr4tch* provides a path toward learning the dynamic world from the scale and diversity of raw video.

In summary, our contributions are:

- We introduce *Scr4tch*, a generalizable model that learns renderable 4D representations entirely from scratch using raw monocular videos, without camera calibration, geometric annotations, or pretrained geometric models.
- We develop a learning strategy that uses motion compression to encourage separation of camera motion and scene dynamics during latent reconstruction, then grounds the learned representations in explicit 4D geometry through joint camera and scene optimization.
- We collect *Casual YouTube*, 2,154 hours of challenging, casually captured videos, for self-supervised learning of scene structure and dynamics at scale without geometric preprocessing.

2 Related Work

Pose-Conditioned Reconstruction. NeRF and 3D Gaussian Splatting (3DGS) optimize implicit volumetric and explicit primitive-based scene representations through differentiable rendering (Mildenhall et al., 2021; Kerbl et al., 2023), with extensive follow-up work on quality, geometry, and efficiency (Barron et al., 2021; Chen et al., 2022; Müller et al., 2022; Yu et al., 2024b; Huang et al., 2024a; Lu et al., 2024). Dynamic extensions model temporal variation through canonical deformation or time-conditioned representations (Pumarola et al., 2021; Park et al., 2021; Fridovich-Keil et al., 2023; Wu et al., 2024; Huang et al., 2024b; Yu et al., 2024a), constrained by pretrained depth and flow priors in monocular settings (Li et al., 2021; Gao et al., 2021; Wang et al., 2023) or by synchronized multi-view capture (Li et al., 2022; Wang et al., 2024a). Feed-forward models remove per-scene optimization by predicting radiance fields, Gaussians, or latent scene representations from posed images (Yu et al., 2021; Chen et al., 2021; Charatan et al., 2024; Chen et al., 2024; Tang et al., 2024; Xu et al., 2024; Zhang et al., 2024; Jin et al., 2025), and have been extended to dynamic scenes using posed monocular videos (Xu et al., 2025), optionally with external depth, flow, and mask supervision (Tian et al., 2023), or synthetic multi-view videos (Ren et al., 2024). All of these methods rely on calibrated cameras.

Pose-Free Reconstruction. Pose-free methods remove camera inputs at inference. Per-scene approaches jointly optimize cameras and radiance fields (Wang et al., 2021; Lin et al., 2021; Jeong et al., 2021), with robust variants adding pretrained depth, correspondence, tracking, or segmentation networks (Bian et al., 2023; Fu et al., 2024; Jiang et al., 2024; Park et al., 2025). Feed-forward models instead learn camera and geometry inference from ground-truth supervision: DUSt3R, MASt3R, and VGGT are trained on 3D-annotated data to predict point maps, cameras, and depths (Wang et al., 2024b; Leroy et al., 2024; Wang et al., 2025a), NoPoSplat removes input poses at inference but requires ground-truth cameras for training (Ye et al., 2024), and D4RT, initialized from a pretrained video encoder, learns cameras and space–time point trajectories under geometric and correspondence supervision (Zhang et al., 2026). Others build on pretrained components, such as MASt3R geometry in Splatt3R, depth and matching modules in PF3plat, and a pretrained geometry backbone with pseudo optical flow in NoPo4D (Smart et al., 2024; Hong et al., 2024a; Balice et al., 2026). Pose-free inference thus does not by itself remove geometric supervision from learning.

Self-Supervised Reconstruction. Self-supervised methods further remove camera and geometric supervision from training. MonoNeRF learns a generalizable radiance field from unlabeled monocular videos (Fu et al., 2023), while FlowCam (Smith et al., 2023) relies on pretrained optical flow,

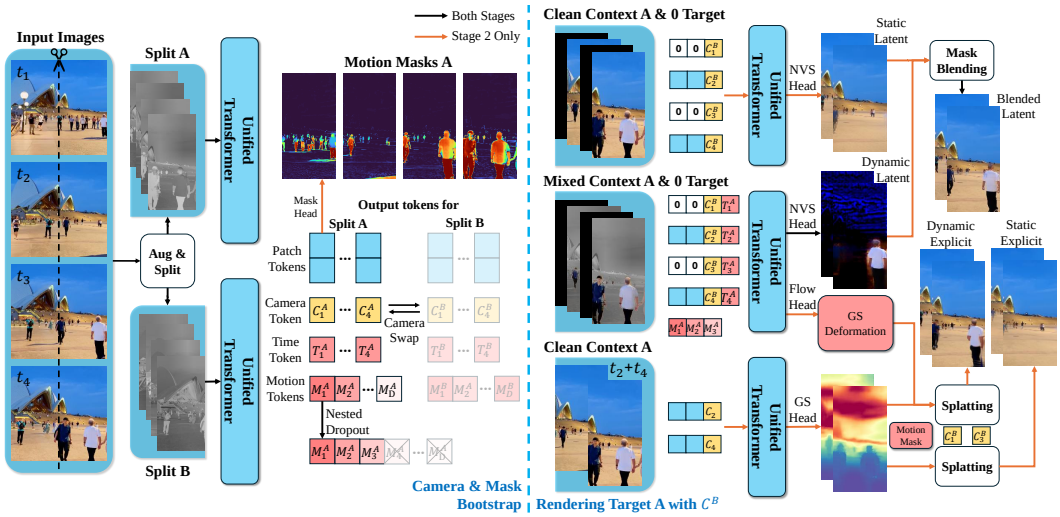


Figure 1: **Pipeline.** Each of the N strongly augmented frames is spatially partitioned into disjoint splits A and B. A bootstrap pass predicts per-view motion masks and camera/time tokens, plus global motion tokens. We exchange camera tokens inferred from complementary splits and apply nested dropout to the motion tokens. Three subsequent passes render target images from static latent features, dynamic latent features, and a deformable Gaussian scene, respectively. Image-space losses on the resulting predictions provide self-supervision.

SelfSplat (Kang et al., 2024) assumes known intrinsics, and UP-LVSM (Wang et al., 2026a) initializes its tokenizer from DINOv2 (Oquab et al., 2024). RayZer factorizes uncalibrated images into predicted cameras and a latent scene through held-out-view reconstruction (Jiang et al., 2025), and XFactor identifies target leakage and context interpolation as shortcuts in such latent-pose systems (Mitchel et al., 2026). Pensieve progresses from latent reconstruction to explicit Gaussian rendering and depth reprojection (Wang et al., 2025b), and E-RayZer and NAS3R learn explicit 3D reconstruction directly from unlabeled multi-view imagery (Zhao et al., 2026; Huang et al., 2026). These methods primarily target static scenes.

Dynamic video introduces a deeper ambiguity, as image motion can arise from either the camera or the scene. Self-supervised monocular depth methods address it through jointly learned motion and masks or mined pseudo-depth (Zhou et al., 2017; Sun & Hariharan, 2023; Nguyen et al., 2024), and EgoMono4D learns from unlabeled egocentric video but starts from pretrained depth and intrinsic estimation (Yuan et al., 2025). RayDer scales self-supervised training to unconstrained dynamic video by absorbing temporal variation into a nuisance state, yet retains static-scene NVS as its output task (Prestel et al., 2026). WildRayZer (Chen et al., 2026) learns motion masks from RayZer and frozen DINOv3 features (Siméoni et al., 2025), but removes transients to render a static background rather than reconstructing their motion. In contrast, we learn from raw monocular RGB videos alone, without calibration, offline pseudo-labels, or pretrained networks, and ground the learned latent representation in explicit 4D geometry with cameras, motion masks, and deformable Gaussians.

3 Proposed Method

Overview. Given N RGB frames $\mathcal{I} = \{I_i\}_{i=1}^{N}$ from an uncalibrated monocular video, *Scr4tch* learns a 4D representation through image reconstruction, without camera or geometric annotations. A shared transformer encodes image patches, per-view camera and time tokens, and global motion tokens. Alternating within- and cross-view attention enables these features to capture spatial structure and cross-view relationships while preserving equivariance to input-view permutations.[1]

As shown in Fig. 1, training progresses from latent reconstruction to explicit geometric alignment. First, a bootstrap pass infers camera and motion conditioning from input frames. Conditioned on these features, the same transformer reconstructs target images from context views, with target im-

[1] Appendix B details the architecture, token organization, and prediction heads.



Figure 2: **Visualizing appearance leakage.** With training augmentation, one retained motion token recovers the moving subject with context-consistent colors. Without augmentation or with 64 tokens, stylized appearance leaks into the reconstruction.

ages zeroed. This stage learns cross-view relationships while restricting appearance leakage and compressing scene dynamics (Sec. 3.1). We then decode cameras and a deformable Gaussian scene, and jointly optimize them through rendering and geometric regularization (Sec. 3.2).

3.1 Latent Reconstruction Pretraining with Motion Token Compression

To stabilize learning from scratch, we first pretrain the transformer with latent reconstruction only, omitting the motion-mask, Gaussian, and deformation heads. A single latent rendering branch reconstructs target images from clean context views and the camera, time, and motion tokens inferred during bootstrapping, with target patch tokens zeroed and no mask-guided input mixing or output blending. Since the bootstrap pass observes the targets, zeroing target patch tokens during rendering does not rule out appearance leakage through the inferred **camera**, **motion**, and **time** tokens.

Camera Token. Appearance leakage through camera tokens has been studied in prior work. RayZer (Jiang et al., 2025) decodes camera tokens into explicit camera parameters, imposing a geometric information bottleneck. During latent pretraining, we instead retain the full latent dimensionality of camera tokens and restrict leakage through cross-split exchange, following XFactor (Mitchel et al., 2026). Since all regions of an image share the same camera, we spatially partition every frame into disjoint splits A and B, bootstrap from both independently, and use the camera tokens from one split to reconstruct the other. No region is thus reconstructed with a camera token that has observed its appearance. As in XFactor, we skip partitioning with probability 0.05 so that the network also learns to bootstrap from full images.

Motion Token. However, motion tokens describe content-specific dynamics and cannot be exchanged across spatial splits like camera tokens, leaving a direct path for target appearance leakage. The capacity needed to represent scene motion varies across videos and is unknown in advance. A fixed budget sufficient for complex motion may therefore leave excess capacity in simpler videos. This excess capacity can transmit target appearance and absorb viewpoint changes even in static regions, offering a shortcut around the constrained camera pathway and causing camera collapse.

Yet, given a reconstructed static scene, a target view can be specified and rendered using just ten parameters of an ideal pinhole camera. This gives camera conditioning a substantial compression advantage for reconstructing large static regions. We exploit this advantage by compressing motion tokens, encouraging a factorization in which cameras explain static content, while motion tokens capture scene dynamics. Specifically, we apply nested dropout to the N_{m} global motion tokens, following FlexTok (Bachmann et al., 2025). During training, we sample ℓ uniformly from $\{0, \ldots, N_{\mathrm{m}}\}$, retain only the first ℓ motion tokens for rendering, and discard the rest. Since camera tokens are always retained while even the first motion token can be dropped, the model is encouraged to prioritize camera conditioning and concentrate essential scene dynamics in the leading motion tokens. When $\ell = 0$, we also remove the per-view time tokens, reducing the same renderer to static latent rendering, similar to dynamic-state dropout in RayDer (Prestel et al., 2026).

However, compression alone does not prevent motion tokens from encoding target appearance. Even a short token sequence can convey the appearance of moving content, allowing the network to reconstruct it without learning its cross-view relationships. We therefore restrict the appearance available during bootstrapping through strong augmentation: grayscale conversion with randomly sampled channel weights, color jitter, and random blurring. These perturbations suppress color and fine appearance cues while retaining structural cues for camera and motion prediction. This encourages motion tokens to encode cross-view relationships of moving content, guiding the recovery of target appearance from clean context views rather than transmitting it directly.

Time Token. Finally, the network needs a per-view cue to retrieve target-specific dynamics from the global motion tokens. With target patch tokens set to zero, camera tokens would otherwise

provide this cue, coupling motion retrieval to camera information. We therefore infer a per-view time token from the input images during bootstrapping, separating the cue for scene dynamics from camera conditioning. To limit appearance leakage through this additional pathway, we constrain each time token to at most one input-dependent degree of freedom.

Figure 2 probes appearance leakage by bootstrapping from stylized versions of all input frames and reconstructing from clean context views. Without appearance augmentation during training, stylized colors leak into the reconstruction even with one motion token. With training augmentation, one token restores the moving subject with context-consistent colors. For the same model, zero tokens reconstruct the static background, whereas 64 tokens partially reintroduce the stylized colors. These observations show that compression and augmentation are complementary: compression alone cannot prevent appearance shortcuts, while excess capacity can still carry target appearance despite augmentation. Without compression, this leakage further collapses the camera pathway (Fig. 5).

We define the reconstruction loss used in both training stages as

$$\mathcal{R}(\widehat{I}, I) = \mathrm{MSE}(\widehat{I}, I) + 0.5\ \mathrm{LPIPS}(\widehat{I}, I). \tag{1}$$

Latent pretraining minimizes $\mathcal{L}^{\mathrm{lat}}_{\mathrm{dyn}} = \mathcal{R}(\widehat{I}^{\mathrm{lat}}_{\mathrm{dyn}}, I)$ as its sole objective, where $\widehat{I}^{\mathrm{lat}}_{\mathrm{dyn}}$ is the dynamic latent reconstruction and I is the corresponding unaugmented RGB image.

3.2 Explicit Reconstruction Alignment

Latent pretraining learns cross-view relationships, but only implicitly: viewpoint and scene dynamics are encoded in latent tokens rather than physical quantities, yielding neither explicit cameras nor scene geometry. Building on these relationships, we decode camera parameters, motion masks, and a deformable Gaussian scene from the learned features, and jointly optimize them through rendering and geometric regularization. This grounds latent reconstruction in physical space, enabling explicit 4D reconstruction and camera-controlled NVS.

Grounding Camera Conditioning. During latent pretraining, viewpoint is specified only by latent camera tokens that are inferred from images and lack physical interpretation. We ground this conditioning in explicit cameras: each bootstrap camera token c_i is decoded into intrinsics and extrinsics $\pi_i = (K_i, R_i, t_i)$, which a shallow MLP maps back to a conditioning token $\widetilde{c}_i = \phi(\pi_i)$. The remapped token replaces c_i in both cross-split exchange and latent rendering. Latent rendering thus becomes controllable by explicit cameras and shares a single set of cameras with Gaussian rendering, jointly refined by reconstruction losses from both pathways. Moreover, since the remapped tokens carry only a few geometric degrees of freedom, appearance leakage remains limited even when bootstrapping from full images without partitioning.

Static and Dynamic Content Separation. Cameras should explain static content, leaving motion tokens only what cameras cannot. To make this division explicit, a mask head decodes bootstrap patch features into a soft motion mask M_i for each view. The same transformer renders each target twice: a static pass conditioned only on cameras, and a dynamic pass that additionally receives motion and time tokens. With view indices omitted, the mask blends the two predictions as

$$\widehat{I}^{\mathrm{lat}}_{\mathrm{blend}} = (1 - M) \odot \widehat{I}^{\mathrm{lat}}_{\mathrm{stat}} + M \odot \widehat{I}^{\mathrm{lat}}_{\mathrm{dyn}}, \tag{2}$$

and $\mathcal{R}(\widehat{I}^{\mathrm{lat}}_{\mathrm{blend}}, I)$ supervises the result. Since the dynamic pass only needs to reconstruct moving content, the mask also mixes clean and augmented context images at its input, exposing clean appearance only in moving regions. Both ends thus push the mask up where motion occurs.

Left unchecked, this pressure admits a degenerate solution: motion tokens, inferred with access to the targets, can substitute for camera tokens, so $M \equiv 1$ explains the entire image through the dynamic pass and collapses the cameras. Mask sparsity and edge-aware smoothness (Ranjan et al., 2019; Lee et al., 2024) counteract this but penalize the mask uniformly, regardless of how well either pass reconstructs a region. We further handicap the dynamic pass within reconstruction itself:

$$\mathcal{A}(\widehat{I}_{\mathrm{stat}}, \widehat{I}_{\mathrm{dyn}}; I, M) = (1 - M) \odot |\widehat{I}_{\mathrm{stat}} - I| + 2M \odot |\widehat{I}_{\mathrm{dyn}} - I|. \tag{3}$$

Raising M at a pixel reduces $\mathcal{A}$ only if its static error exceeds twice its dynamic error, biasing the mask toward static explanations unless the dynamic pass is substantially better. Since a thin static

structure, once marked as moving, receives little static gradient under $(1 - M)$, we also apply $\mathcal{R}$ to the full static prediction, yielding the latent objective

$$\mathcal{L}_{\text{lat}} = \mathcal{R}(\widehat{I}^{\text{lat}}_{\text{blend}}, I) + \mathcal{R}(\widehat{I}^{\text{lat}}_{\text{stat}}, I) + \lambda_{\mathcal{A}}\, \mathcal{A}(\widehat{I}^{\text{lat}}_{\text{stat}}, \widehat{I}^{\text{lat}}_{\text{dyn}}; I, M), \tag{4}$$

where $\lambda_{\mathcal{A}}$ weights the asymmetric cost.

Deformable Gaussian Scene Construction. Decoded cameras and masks alone do not tie the learned features to physical space, since latent rendering remains a learned image synthesizer. We therefore introduce a deformable Gaussian scene as the bridge: its rendering follows projective geometry, so reconstructing the targets requires consistent cameras, depths, and motion. A separate pass of the same transformer predicts pixel-aligned base Gaussians (Kerbl et al., 2023) from clean context images only. We keep target views out of this pass because jointly estimated camera conditioning can convey target-view cues, such as the absence of viewpoint change, and permit reconstruction without meaningful depth (Fig. 3). Each Gaussian center is placed by backprojecting the predicted depth D_i with the decoded camera π_i. For dynamics, we deform the base Gaussians toward each target time (Wu et al., 2024): a lightweight head modulates context features from dynamic latent rendering with the target time token through FiLM and predicts 7-DoF residuals in image-plane position, depth, rotation, and opacity. Gated by the motion mask, these deformations yield scene states at multiple target times without additional transformer passes.

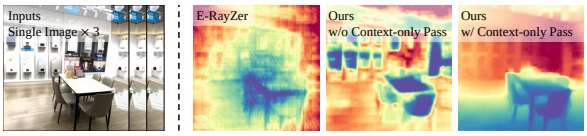


Figure 3: **Effect of the context-only pass.** By keeping target views out, it prevents leaking the absence of viewpoint change among repeated inputs and yields meaningful depth.

Explicit rendering follows the structure of the latent objective, adapted to Gaussians. Since deformed Gaussians render complete target images $\widehat{I}^{\text{exp}}_{\text{dyn}}$, no blending is needed. The static rendering $\widehat{I}^{\text{exp}}_{\text{stat}}$ uses undeformed Gaussians with opacity attenuated by the motion mask, so moving objects do not corrupt the static rendering. Supervising the full static image still trains static structures misclassified as moving, but also forces undeformed Gaussians to fit moving objects, which can distort the cameras. We therefore decouple the two effects: with equal probability, the static rendering is supervised either over the full image with detached cameras ($W = 1$), or only over static regions with camera gradients enabled ($W = 1 - M$). We term this camera gradient routing (CGR): static rendering passes camera gradients only through static regions. The explicit objective is

$$\mathcal{L}_{\text{exp}} = \mathcal{R}(\widehat{I}^{\text{exp}}_{\text{dyn}}, I) + \mathcal{R}_W(\widehat{I}^{\text{exp}}_{\text{stat}}, I) + \lambda_{\mathcal{A}}\, \mathcal{A}(\widehat{I}^{\text{exp}}_{\text{stat}}, \widehat{I}^{\text{exp}}_{\text{dyn}}; I, M), \tag{5}$$

where $\mathcal{R}_W$ is the W-weighted form of $\mathcal{R}$, and $\mathcal{A}$ again favors static explanations.

Geometric Regularization beyond Rendering. Rendering constrains geometry only through appearance, so we add self-supervised geometric objectives. Predicted depths D_i receive only indirect gradients through Gaussian splatting and retain a patch-grid pattern even after the rendered context depths D^{render}_i converge. We therefore distill D^{render}_i into D_i (Fig. 9), transferring geometric detail into the shared features. Photometric reprojection further ties each context depth to neighboring frames: points backprojected from D_i are deformed and projected into the two nearest input frames to sample RGB, supplying gradients to depths, cameras, and deformations. Taking the minimum error over the two frames reduces sensitivity to occlusion (Godard et al., 2019).

Depth and object motion can nevertheless compensate for one another in moving regions, where a wrong depth with compensating motion reconstructs the images equally well (Nguyen et al., 2024). Single-view depth is free of this ambiguity: from a single frame, the network cannot tell whether an object moves and must rely on monocular structural priors. Motivated by this observation, we apply the same network to each frame independently to predict D^{single}_i and let the motion mask route mutual supervision: D^{single}_i supervises D^{render}_i in moving regions, while D^{render}_i supervises D^{single}_i in static regions, with each teacher detached. All objectives jointly train the shared network using only RGB observations and its own predictions; Appendices C.3 and C.4 detail the depth losses and combined objective.

Table 1: Camera estimation on Wild-SLAM Mocap (dynamic) / DL3DV-Evaluation (static). **Bold** and underline mark the best and second-best results among self-supervised methods. SelfSplat and PF3plat take ground-truth intrinsics as input; PF3plat and EgoMono4D build on pretrained networks.

Method	Resolution	ARE_f ↓	RRA@5 ↑	RRA@15 ↑	RRA@30 ↑	RTA@5 ↑	RTA@15 ↑	RTA@30 ↑
Supervised Methods								
MonST3R	288 × 512	0.062 / 0.294	0.917 / 0.962	0.984 / 0.989	1.000 / 0.993	0.357 / 0.494	0.745 / 0.915	0.869 / 0.967
NeoVerse	336 × 560	0.020 / 0.030	0.987 / 0.948	1.000 / 0.993	1.000 / 0.998	0.705 / 0.670	0.936 / 0.903	0.987 / 0.952
Self-supervised Methods								
SelfSplat	256 × 448	N/A	0.957 / 0.633	0.997 / 0.813	**1.000** / 0.922	0.153 / 0.183	0.539 / 0.477	0.753 / 0.653
PF3plat	256 × 256	N/A	0.946 / 0.891	0.973 / 0.979	**1.000** / 0.988	0.284 / 0.351	0.890 / 0.859	0.949 / 0.954
NAS3R	256 × 256	0.106 / 0.264	0.858 / 0.832	0.979 / 0.932	0.997 / 0.966	0.131 / 0.417	0.515 / 0.774	0.745 / 0.886
E-RayZer	256 × 256	0.121 / **0.068**	0.957 / 0.923	**1.000** / 0.970	**1.000** / 0.985	0.319 / 0.771	0.684 / 0.913	0.839 / 0.955
EgoMono4D	288 × 384	0.052 / 0.129	0.962 / 0.795	0.997 / 0.956	**1.000** / 0.985	0.035 / 0.054	0.247 / 0.322	0.587 / 0.641
Ours	256 × 256	**0.040** / 0.134	**1.000 / 0.931**	**1.000 / 0.980**	**1.000 / 0.991**	**0.528 / 0.789**	**0.930 / 0.948**	**0.992 / 0.969**

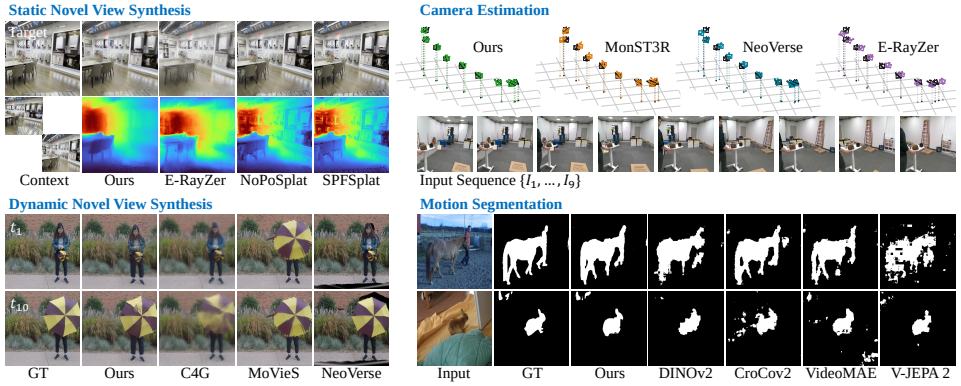


Figure 4: **Qualitative comparisons.** Top left: in static NVS, *Scr4tch* renders images and depths well aligned with the scene content, whereas other methods produce texture-distorted depths. Top right: in camera estimation, the trajectory of *Scr4tch* aligns best with the ground truth despite a cart moving with the camera, recovering true camera motion undisturbed by moving content. Bottom left: in dynamic NVS, *Scr4tch* renders the correct scene state at each time from novel viewpoints, disentangling time from camera motion. Bottom right: in motion segmentation probing, *Scr4tch* yields the cleanest masks and even captures small, distant moving objects.

4 Experiments

4.1 Datasets and Implementation Details

To learn from challenging, casually captured videos, we collect *Casual YouTube*, 2,154 hours of monocular footage covering walking, cycling, and first-person driving under sunny, rainy, snowy, and nighttime conditions. For scene diversity, we mix it with SpatialVID-HQ (Wang et al., 2026b), RealEstate10K (Zhou et al., 2018), and DL3DV-10K (Ling et al., 2024), sampling the four sources with probabilities of 50%, 30%, 10%, and 10%. Only RGB frames are used from all sources, without camera or geometric annotations. Each training example contains 2 to 9 frames. Context and target indices are sampled randomly during latent pretraining and interleaved temporally during explicit alignment, so that targets remain well covered by the context-predicted Gaussians, avoiding large unseen regions that would destabilize training. We train and evaluate *Scr4tch* at 256^2 resolution. Further details are provided in Appendix D.

4.2 Comparisons with Existing Methods

Camera Estimation. We evaluate on Wild-SLAM Mocap (Zheng et al., 2025) (dynamic) and DL3DV-Evaluation (Ling et al., 2024) (static), running each method on the full sequence with its preferred window length and resolution, and report relative rotation and translation accuracy (RRA/RTA) over all adjacent frame pairs and the relative focal-length error ARE_f (Table 1). Without camera labels or pretrained networks, *Scr4tch* achieves the best accuracy among self-supervised methods, including those built on pretrained networks, except for a higher focal error on static scenes. On dynamic scenes, it also surpasses MonST3R (Zhang et al., 2025), fine-tuned from DUSt3R (Wang et al., 2024b) with geometric supervision and run at higher resolution, and remains

competitive with the supervised NeoVerse (Yang et al., 2026). We attribute this robustness to separating camera motion from scene dynamics: *Scr4tch* recovers poses aligned with the ground truth even when a cart moves along with the camera and stays nearly fixed in the image (Fig. 4). Modeling dynamics comes at no cost on static scenes: although designed for dynamic reconstruction, *Scr4tch* surpasses the static-scene state of the art E-RayZer (Zhao et al., 2026) in pose accuracy and attains the highest RTA among all methods.

Novel View Synthesis. We evaluate NVS on both static and dynamic scenes (Table 2). Following prior protocols (Hong et al., 2024b; Wang et al., 2025b), target images are excluded from the input, so the evaluation measures synthesis of unseen views rather than target-image autoencoding. Ground-truth target poses are aligned to each prediction via the inferred context cameras, except for MVSplat (Chen et al., 2024) and MoVieS (Lin et al., 2026), which take ground-truth cameras as input. On static scenes from DL3DV-Evaluation (Ling et al., 2024), latent rendering of *Scr4tch* (Ours$^{\text{lat}}$) outperforms all methods, including NoPoSplat (Ye et al., 2024) trained with ground-truth cameras and SPFSplat (Huang & Mikolajczyk, 2025) initialized from MASt3R (Leroy et al., 2024), and Gaussian rendering (Ours$^{\text{exp}}$) also exceeds all baselines in PSNR, with depths that follow scene structure rather than texture (Fig. 4). On dynamic scenes from NVIDIA Dynamic (Yoon et al., 2020), each of the 12 input images comes from a different camera at a different time, and we render all other cameras at each input time by deforming the Gaussians with the time token of that image. We report only Gaussian rendering here, as latent rendering may be subject to information leakage. Without the external geometric and motion supervision used by NeoVerse (Yang et al., 2026), C4G (Kim et al., 2026), and MoVieS, and despite training on at most 9 frames, *Scr4tch* achieves the highest PSNR and renders the correct scene state at each time (Fig. 4), indicating that time and viewpoint are disentangled in the learned representation (Fig. 7).

Table 2: Novel view synthesis.

Method	PSNR ↑	SSIM ↑	LPIPS ↓
Static Scene			
MVSplat	20.15	0.642	0.215
NoPoSplat	20.87	<u>0.669</u>	<u>0.197</u>
SPFSplat	20.58	0.654	0.213
SelfSplat	16.20	0.406	0.496
NAS3R	18.23	0.515	0.294
E-RayZer	18.77	0.546	0.318
Ours$^{\text{exp}}$	<u>21.32</u>	0.666	0.201
Ours$^{\text{lat}}$	**21.70**	**0.674**	**0.177**
Dynamic Scene			
NeoVerse	15.42	0.316	<u>0.369</u>
C4G	16.83	0.295	0.545
MoVieS	<u>17.71</u>	**0.369**	**0.354**
Ours$^{\text{exp}}$	**17.80**	<u>0.345</u>	0.394

Depth Probing. Beyond reconstruction, we examine whether the learned representation itself encodes scene geometry by probing frozen backbones for multi-view depth estimation on BlendedMVS (Yao et al., 2020) with 3 input frames and for single-view depth estimation on NYUv2 (Silberman et al., 2012) (Table 3). For all methods, we remove the original output heads and train only the same lightweight head, two convolutional residual blocks on linearly projected backbone features, with ground-truth depth (Appendix D); DINOv2 (Oquab et al., 2024) serves as a general-purpose reference that processes each frame independently. *Scr4tch* achieves the best multi-view depth, and all methods with explicit 3D reconstruction outperform the latent-only RayZer, XFactor, and RayDer, consistent with our motivation for grounding latent representations in explicit geometry. On single-view depth, *Scr4tch* trails only DINOv2 and SelfSplat, both built on large-scale pretrained image encoders, and stays well ahead of the other self-supervised reconstruction methods, showing that geometry learned from raw video alone remains informative without cross-view cues.

Table 3: Depth probing (multi-view / single-view).

Method	ARE ↓	$\delta < 1.25$ ↑
DINOv2	0.097 / 0.093	0.906 / 0.913
RayZer	0.143 / 0.228	0.812 / 0.586
XFactor	0.122 / 0.196	0.861 / 0.652
RayDer	0.123 / 0.169	0.841 / 0.737
SelfSplat	0.086 / **0.077**	0.929 / **0.931**
NAS3R	0.103 / 0.181	0.896 / 0.693
E-RayZer	0.059 / 0.176	0.957 / 0.705
Ours	**0.056** / 0.117	**0.966** / 0.860

Motion Segmentation Probing. We further probe whether the representation distinguishes independently moving content from camera motion through moving object segmentation on FBMS-59 (Ochs et al., 2014), where all moving objects in a frame form the foreground. All methods take two frames as input, with DINOv2 again processing each frame independently, and the same probing head is trained on sparse annotations of only 353 frames and evaluated on 367 held-out frames (Table 4). With these sparse labels, *Scr4tch* outperforms image, two-view, and video self-supervised representations, namely DINOv2, CroCo v2 (Weinzaepfel et al., 2023), VideoMAE (Tong et al., 2022), and V-JEPA 2 (Assran et al., 2025), by a large margin, and even segments small, distant moving objects (Fig. 4). Notably, the video models fall below the per-frame DINOv2, suggesting that generic spatiotemporal features do not readily separate object motion from camera motion, whereas *Scr4tch* is trained to explain static content through cameras and only the remaining dynamics through motion tokens.

Table 4: Motion segmentation.

Method	IoU ↑	P ↑	R ↑
DINOv2	0.665	0.814	0.784
CroCov2	0.649	0.789	0.785
VideoMAE	0.612	0.767	0.752
V-JEPA 2	0.562	0.747	0.694
Ours	**0.766**	**0.895**	**0.842**

Table 5: Ablations on Wild-SLAM Mocap and NVIDIA Dynamic.

Setting	ARE_f ↓	RRA@5 ↑	RTA@5 ↑	PSNR ↑
$\mathcal{L}_{\text{lat}} + \mathcal{L}_{\text{exp}}$	0.071	0.820	0.478	17.48
$\mathcal{L}_{\text{lat}} + \mathcal{L}_{\text{exp}} + \mathcal{L}_{\text{sample}}$	0.060	0.834	0.512	17.51
$\mathcal{L}_{\text{lat}} + \mathcal{L}_{\text{exp}} + \mathcal{L}_{\text{sample}} + \mathcal{L}_{\text{sv}}$	0.056	0.903	0.513	17.85
$\mathcal{L}_{\text{lat}} + \mathcal{L}_{\text{exp}} + \mathcal{L}_{\text{sample}} + \mathcal{L}_{\text{sv}}$ + CGR	**0.045**	**0.906**	**0.592**	**17.87**
w/o *Casual YouTube*	0.098	0.863	0.493	17.77
w/o nested dropout	0.653	0.174	0.011	16.67

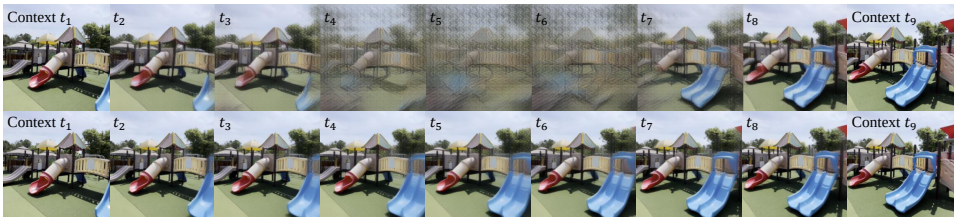


Figure 5: **Effect of motion token compression on camera control.** Latent renderings without motion tokens along cameras interpolated between two context views (t_1 and t_9). Without nested dropout (top), the renderings do not follow the camera and only blend the two contexts; with it (bottom), scene content moves consistently with the interpolated viewpoints.

4.3 Ablation Studies

We first examine motion token compression in latent pretraining. When bootstrapping from stylized frames and rendering from clean contexts (Fig. 2), a single motion token restores the moving subject with colors consistent with the clean contexts rather than the stylized targets. The motion token thus carries cross-view relationships of moving content instead of encoding target appearance, which requires both compression and augmentation: without augmentation, even one token leaks stylized colors, and 64 tokens also reintroduce them. Fig. 5 further tests viewpoint control by rendering intermediate views without motion tokens. With nested dropout, scene content moves consistently with the viewpoint; without it, the renderings ignore the camera and only blend the two contexts. Uncompressed motion tokens carry target appearance during training, so reconstruction retrieves targets from them instead of predicting targets from context views and cameras, and the camera pathway collapses: RTA@5 falls to 0.011 (Table 5), while PSNR remains moderate mainly owing to large background regions. Restricting motion capacity is therefore key to camera-controlled NVS.

Table 5 evaluates the alignment objectives on dynamic camera estimation on Wild-SLAM Mocap and dynamic NVS on NVIDIA Dynamic, adding components cumulatively. Photometric reprojection $\mathcal{L}_{\text{sample}}$ and CGR both improve pose accuracy, and CGR brings the largest gain in translation (Fig. 10), confirming that it keeps moving objects from distorting the cameras. Single-view and multi-view depth consistency $\mathcal{L}_{\text{sv}}$ brings the largest rendering gain: under reconstruction alone, moving objects can take incorrect depths and render wrongly from other viewpoints, whereas single-view depth corrects them (Fig. 11). Finally, removing *Casual YouTube* from training clearly degrades camera estimation. These videos are used as raw footage without any geometric preprocessing, yet they substantially improve performance, indicating that *Scr4tch* can exploit unprocessed Internet video at scale.

5 Conclusion

We presented *Scr4tch*, a model that learns renderable 4D representations entirely from scratch using raw monocular videos, without camera calibration, geometric annotations, or pretrained geometric models. By restricting motion capacity during latent reconstruction, *Scr4tch* separates camera motion from scene dynamics, and then grounds the learned representations in explicit geometry through joint camera and scene optimization. The resulting model estimates accurate cameras in both dynamic and static scenes, supports camera-controlled NVS, and learns features from which geometry and independent motion are readily decoded. Its performance further improves with unprocessed Internet video, suggesting a path toward learning the structure and dynamics of the world from raw video at scale. As a **limitation**, constrained by training scale and the difficulty of optimizing explicit representations, *Scr4tch* still trails large reconstruction foundation models, and closing this gap is left for future work.

AI USE STATEMENT

We used large language models (LLMs) to polish the writing of this paper, including wording, clarity, and presentation, and to write non-core code. All LLM-generated content was reviewed and verified by the authors, who take full responsibility for the content of this paper.

ETHICS STATEMENT

Our *Casual YouTube* collection consists only of publicly accessible YouTube videos. We use the videos solely for research purposes. The collected videos will not be redistributed; to support reproducibility, we will release only the video URLs. Moreover, *Scr4tch* is a reconstruction model without generative capability: it re-renders content observed in the given context views and does not generate faces or other personal information absent from them.

REFERENCES

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, et al. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025. URL https://arxiv.org/abs/2506.09985. 8

Roman Bachmann, Jesse Allardice, David Mizrahi, Enrico Fini, Oğuzhan Fatih Kar, Elmira Amirloo, Alaaeldin El-Nouby, Amir Zamir, and Afshin Dehghan. FlexTok: Resampling images into 1D token sequences of flexible length. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 2241–2292. PMLR, 2025. URL https://proceedings.mlr.press/v267/bachmann25a.html. 2, 4

Matteo Balice, Yanik Kunzi, Chenyangguang Zhang, Matteo Matteucci, Marc Pollefeys, and Sunghwan Hong. No pose, no problem in 4d: Feed-forward dynamic gaussians from unposed multi-view videos. *arXiv preprint arXiv:2605.22190*, 2026. 1, 2

Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5855–5864, 2021. 2

Wenjing Bian, Zirui Wang, Kejie Li, Jia-Wang Bian, and Victor Adrian Prisacariu. Nope-nerf: Optimising neural radiance field with no pose prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4160–4169, 2023. 2

David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19457–19467, 2024. 2

Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 14124–14133, 2021. 2

Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *European conference on computer vision*, pp. 333–350. Springer, 2022. 2

Xuweiyi Chen, Wentao Zhou, and Zezhou Cheng. WildRayZer: Self-supervised large view synthesis in dynamic environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1252–1264, 2026. 1, 3

Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In *European Conference on Computer Vision*, pp. 370–386. Springer, 2024. 2, 8

Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12479–12488, 2023. 2

Yang Fu, Ishan Misra, and Xiaolong Wang. MonoNeRF: Learning generalizable NeRFs from monocular videos without camera poses. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10392–10404, 2023. 2

Yang Fu, Sifei Liu, Amey Kulkarni, Jan Kautz, Alexei A Efros, and Xiaolong Wang. Colmap-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20796–20805, 2024. 2

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*, pp. 5712–5721, 2021. 2

Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3828–3838, 2019. 6, 22

Sunghwan Hong, Jaewoo Jung, Heeseong Shin, Jisang Han, Jiaolong Yang, Chong Luo, and Seungryong Kim. Pf3plat: Pose-free feed-forward 3d gaussian splatting. *arXiv preprint arXiv:2410.22128*, 2024a. 2

Sunghwan Hong, Jaewoo Jung, Heeseong Shin, Jiaolong Yang, Seungryong Kim, and Chong Luo. Unifying correspondence pose and nerf for generalized pose-free novel view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20196–20206, 2024b. 8

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 conference papers*, pp. 1–11, 2024a. 2

Ranran Huang and Krystian Mikolajczyk. No pose at all: Self-supervised pose-free 3d gaussian splatting from sparse views. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27947–27957, 2025. 8

Ranran Huang, Weixun Luo, Ye Mao, and Krystian Mikolajczyk. From none to all: Self-supervised 3D reconstruction via novel view synthesis. *arXiv preprint arXiv:2603.27455*, 2026. URL https://arxiv.org/abs/2603.27455. 3

Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4220–4230, 2024b. 2

Yoonwoo Jeong, Seokjun Ahn, Christopher Choy, Anima Anandkumar, Minsu Cho, and Jaesik Park. Self-calibrating neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5846–5854, 2021. 2

Hanwen Jiang, Hao Tan, Peng Wang, Haian Jin, Yue Zhao, Sai Bi, Kai Zhang, Fujun Luan, Kalyan Sunkavalli, Qixing Huang, and Georgios Pavlakos. RayZer: A self-supervised large view synthesis model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4918–4929, 2025. 1, 3, 4

Kaiwen Jiang, Yang Fu, Mukund Varma T, Yash Belhe, Xiaolong Wang, Hao Su, and Ravi Ramamoorthi. A construct-optimize approach to sparse view synthesis without camera pose. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024. 2

Haian Jin, Hanwen Jiang, Hao Tan, Kai Zhang, Sai Bi, Tianyuan Zhang, Fujun Luan, Noah Snavely, and Zexiang Xu. LVSM: A large view synthesis model with minimal 3D inductive bias. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=QQBPWtvtcn. 2

Gyeongjin Kang, Jisang Yoo, Jihyeon Park, Seungtae Nam, Hyeonsoo Im, Sangheon Shin, Sangpil Kim, and Eunbyung Park. Selfsplat: Pose-free and 3d prior-free generalizable 3d gaussian splatting. *arXiv preprint arXiv:2411.17190*, 2024. 3

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139:1–139:14, 2023. doi: 10.1145/3592433. 2, 6

Mungyeom Kim, Minkyeong Jeon, Honggyu An, Jaewoo Jung, Hyuna Ko, Jisang Han, Hyeonseo Yu, Donghwan Shin, Sunghwan Hong, Takuya Narihira, Kazumi Fukuda, Yuki Mitsufuji, and Seungryong Kim. Learning global motion with compact gaussians for feed-forward 4D reconstruction. *arXiv preprint arXiv:2605.31595*, 2026. URL https://arxiv.org/abs/2605.31595. 8

Yao-Chih Lee, Zhoutong Zhang, Kevin Blackburn-Matzen, Simon Niklaus, Jianming Zhang, Jia-Bin Huang, and Feng Liu. Fast view synthesis of casual videos with soup-of-planes. In *European Conference on Computer Vision*, 2024. URL https://casual-fvs.github.io/. 5, 22

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European Conference on Computer Vision*, pp. 71–91. Springer, 2024. 2, 8

Tianye Li, Mira Slavcheva, Michael Zollhöfer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, and Zhaoyang Lv. Neural 3D video synthesis from multi-view video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5521–5531, 2022. 2

Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6498–6508, 2021. 2

Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5741–5751, 2021. 2

Chenguo Lin, Yuchen Lin, Panwang Pan, Yifan Yu, Tao Hu, Honglei Yan, Katerina Fragkiadaki, and Yadong Mu. MoVieS: Motion-aware 4D dynamic view synthesis in one second. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 295–306, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Lin_MoVieS_Motion-Aware_4D_Dynamic_View_Synthesis_in_One_Second_CVPR_2026_paper.html. 8

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22160–22169, 2024. 7, 8

Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the CoordConv solution. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/60106888f8977b71e1f15db7bc9a88d1-Abstract.html. 20

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 23

Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20654–20664, 2024. 2

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

Thomas W. Mitchel, Hyunwoo Ryu, and Vincent Sitzmann. True self-supervised novel view synthesis is transferable. In *International Conference on Learning Representations*, 2026. 3, 4

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)*, 41(4):1–15, 2022. 2

Hoang Chuong Nguyen, Tianyu Wang, Jose M. Alvarez, and Miaomiao Liu. Mining supervision for dynamic regions in self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10446–10455, 2024. 3, 6

Peter Ochs, Jitendra Malik, and Thomas Brox. Segmentation of moving objects by long term video analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(6):1187–1200, 2014. 8

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=a68SUt6zFt. 3, 8

Jongmin Park, Minh-Quan Viet Bui, Juan Luis Gonzalez Bello, Jaeho Moon, Jihyong Oh, and Munchurl Kim. Splinegs: Robust motion-adaptive spline for real-time dynamic 3d gaussians from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26866–26875, 2025. 2

Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5865–5874, 2021. 2

Ulrich Prestel, Stefan Andreas Baumann, Nick Stracke, and Björn Ommer. RayDer: Scalable self-supervised novel view synthesis from real-world video. *arXiv preprint arXiv:2605.31535*, 2026. 1, 3, 4, 17

Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-NeRF: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10318–10327, 2021. 2

Anurag Ranjan, Varun Jampani, Lukas Balles, Kihwan Kim, Deqing Sun, Jonas Wulff, and Michael J. Black. Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12240–12249, 2019. URL https://github.com/anuragranj/cc. 5, 22

Jiawei Ren, Kevin Xie, Ashkan Mirzaei, Hanxue Liang, Xiaohui Zeng, Karsten Kreis, Ziwei Liu, Antonio Torralba, Sanja Fidler, Seung Wook Kim, and Huan Ling. L4GM: Large 4d gaussian reconstruction model. In *Advances in Neural Information Processing Systems*, volume 37, 2024. 1, 2

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *European Conference on Computer Vision*, pp. 746–760, 2012. 8

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025. URL https://arxiv.org/abs/2508.10104. 3

Brandon Smart, Chuanxia Zheng, Iro Laina, and Victor Adrian Prisacariu. Splatt3r: Zero-shot gaussian splatting from uncalibrated image pairs. *arXiv preprint arXiv:2408.13912*, 2024. 2

Cameron Smith, Yilun Du, Ayush Tewari, and Vincent Sitzmann. Flowcam: Training generalizable 3d radiance fields without camera poses via pixel-aligned scene flow. *arXiv preprint arXiv:2306.00180*, 2023. 2

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021. URL https://arxiv.org/abs/2104.09864. 20

Yihong Sun and Bharath Hariharan. Dynamo-Depth: Fixing unsupervised depth estimation for dynamical scenes. In *Advances in Neural Information Processing Systems*, volume 36, 2023. 3

Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pp. 1–18. Springer, 2024. 2

Fengrui Tian, Shaoyi Du, and Yueqi Duan. Mononerf: Learning a generalizable dynamic radiance field from monocular videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17903–17913, 2023. 2

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Advances in Neural Information Processing Systems*, volume 35, 2022. 8

Chaoyang Wang, Lachlan Ewen MacDonald, László A. Jeni, and Simon Lucey. Flow supervision for deformable NeRF. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21128–21137, 2023. 2

Haoru Wang, Kai Ye, Minghan Qin, Yangyan Li, Wenzheng Chen, and Baoquan Chen. The less you depend, the more you learn: Synthesizing novel views from sparse, unposed images without any 3d knowledge. In *International Conference on Learning Representations*, volume 2026, pp. 41717–41739, 2026a. 3

Jiahao Wang, Yufeng Yuan, Rujie Zheng, Youtian Lin, Jian Gao, Lin-Zhuo Chen, Yajie Bao, Chang Zeng, Yanxi Zhou, Xiao-Xiao Long, et al. Spatialvid: A large-scale video dataset with spatial annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42592–42603, 2026b. 7

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5294–5306, 2025a. 2, 20

Ruoyu Wang, Yi Ma, and Shenghua Gao. Recollection from pensieve: Novel view synthesis via learning from uncalibrated videos. *arXiv preprint arXiv:2505.13440*, 2025b. 1, 3, 8

Shuo Wang, Binbin Huang, Ruoyu Wang, and Shenghua Gao. Space-time 2D gaussian splatting for accurate surface reconstruction under complex dynamic scenes. *arXiv preprint arXiv:2409.18852*, 2024a. 2

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20697–20709, 2024b. 2, 7

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. In *International Conference on Learning Representations*, 2026c. URL https://arxiv.org/abs/2507.13347. 21

Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021. 2

Philippe Weinzaepfel, Thomas Lucas, Vincent Leroy, Yohann Cabon, Vaibhav Arora, Romain Brégier, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jérôme Revaud. CroCo v2: Improved cross-view completion pre-training for stereo matching and optical flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17969–17980, 2023. 8

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*, pp. 20310–20320, 2024. 2, 6

Yinghao Xu, Zifan Shi, Wang Yifan, Hansheng Chen, Ceyuan Yang, Sida Peng, Yujun Shen, and Gordon Wetzstein. Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation. In *European Conference on Computer Vision*, pp. 1–20. Springer, 2024. 2

Zhen Xu, Zhengqin Li, Zhao Dong, Xiaowei Zhou, Richard Newcombe, and Zhaoyang Lv. 4DGT: Learning a 4D gaussian transformer using real-world monocular videos. In *Advances in Neural Information Processing Systems*, volume 38, 2025. 1, 2

Yuxue Yang, Lue Fan, Ziqi Shi, Junran Peng, Feng Wang, and Zhaoxiang Zhang. NeoVerse: Enhancing 4D world model with in-the-wild monocular videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://arxiv.org/abs/2601.00393. 8

Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. BlendedMVS: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1790–1799, 2020. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Yao_BlendedMVS_A_Large-Scale_Dataset_for_Generalized_Multi-View_Stereo_Networks_CVPR_2020_paper.html. 8

Botao Ye, Sifei Liu, Haofei Xu, Xueting Li, Marc Pollefeys, Ming-Hsuan Yang, and Songyou Peng. No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. *arXiv preprint arXiv:2410.24207*, 2024. 2, 8

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*, pp. 5336–5345, 2020. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Yoon_Novel_View_Synthesis_of_Dynamic_Scenes_With_Globally_Coherent_Depths_CVPR_2020_paper.html. 8

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4578–4587, 2021. 2

Heng Yu, Joel Julin, Zoltán Á Milacski, Koichiro Niinuma, and László A Jeni. Cogs: Controllable gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21624–21633, 2024a. 2

Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19447–19456, 2024b. 2

Chengbo Yuan, Geng Chen, Li Yi, and Yang Gao. Self-supervised monocular 4D scene reconstruction for egocentric videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8863–8874, 2025. 1, 3

Chuhan Zhang, Guillaume Le Moing, Skanda Koppula, Ignacio Rocco, Liliane Momeni, Junyu Xie, Shuyang Sun, Rahul Sukthankar, Joëlle K Barral, Raia Hadsell, et al. Efficiently reconstructing dynamic scenes one d4rt at a time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7382–7392, 2026. 2

Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. MonST3R: A simple approach for estimating geometry in the presence of motion. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2410.03825. 7

Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. In *European Conference on Computer Vision*, pp. 1–19. Springer, 2024. 2

Qitao Zhao, Hao Tan, Qianqian Wang, Sai Bi, Kai Zhang, Kalyan Sunkavalli, Shubham Tulsiani, and Hanwen Jiang. E-RayZer: Self-supervised 3D reconstruction as spatial visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7525–7535, 2026. 3, 8

Jianhao Zheng, Zihan Zhu, Valentin Bieri, Marc Pollefeys, Songyou Peng, and Iro Armeni. WildGS-SLAM: Monocular gaussian splatting SLAM in dynamic environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11461–11471, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Zheng_WildGS-SLAM_Monocular_Gaussian_Splatting_SLAM_in_Dynamic_Environments_CVPR_2025_paper.html. 7

Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1851–1858, 2017. 3

Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 37, 2018. URL https://arxiv.org/abs/1805.09817. 7

A Additional Results



Figure 6: **Camera estimation under out-of-distribution appearance.** Given a real frame (left) and a stylized frame (right), *Scr4tch* recovers the relative camera between them (middle), consistent with the ground truth, despite the drastic appearance change.

A.1 Out-of-Distribution Appearance

Figure 6 pairs a real frame with a stylized frame of the same scene, whose appearance lies far outside the training distribution. *Scr4tch* still recovers the relative camera between the two views in agreement with the ground truth, even though the person also moves between them. This suggests that camera estimation relies on scene structure rather than on appearance statistics seen during training, consistent with bootstrapping from strongly augmented frames, which suppresses color and fine appearance cues.

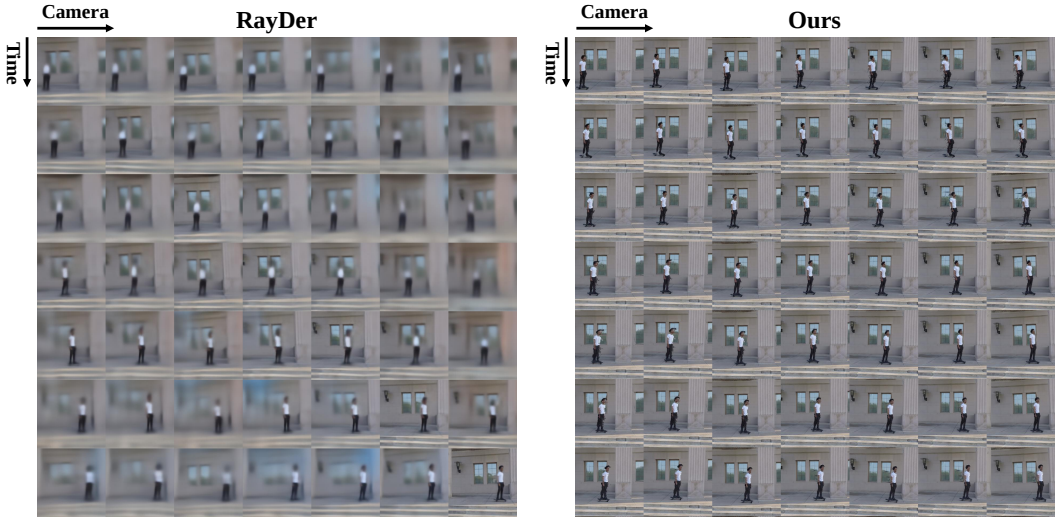


Figure 7: **Time-view matrices on NVIDIA Dynamic.** Both methods are trained self-supervised on dynamic videos. Each column uses the camera of one view, and each row the dynamic condition of one time. In RayDer, the viewpoint barely follows the camera along each row but shifts with the dynamic state down each column, entangling camera motion with scene dynamics. *Scr4tch* disentangles them: the camera controls the viewpoint, and the time token controls scene motion.

A.2 Disentangling Camera and Scene Motion

RayDer (Prestel et al., 2026) and *Scr4tch* are both trained self-supervised on dynamic videos, yet only *Scr4tch* separates camera motion from scene dynamics. Figure 7 shows this through time-view matrices on NVIDIA Dynamic, extending the dynamic NVS evaluation in Table 2. RayDer conditions latent rendering on a latent camera and a dynamic state; we first infer both for every input

view and then, keeping the context views fixed, render with the camera of view i and the dynamic state of view j. The resulting renderings barely follow changes in the camera, whereas changing the dynamic state moves both the viewpoint and the scene content: viewpoint changes are absorbed into the dynamic state rather than explained by the camera. For *Scr4tch*, the Gaussians predicted from the context views are deformed with the time token of view j and rendered with the decoded camera of view i. The camera alone changes the viewpoint, and the time token alone changes the scene state. Self-supervision on dynamic videos thus suffices to disentangle camera motion from scene dynamics, and the explicit 4D scene makes every combination of viewpoint and time directly renderable.

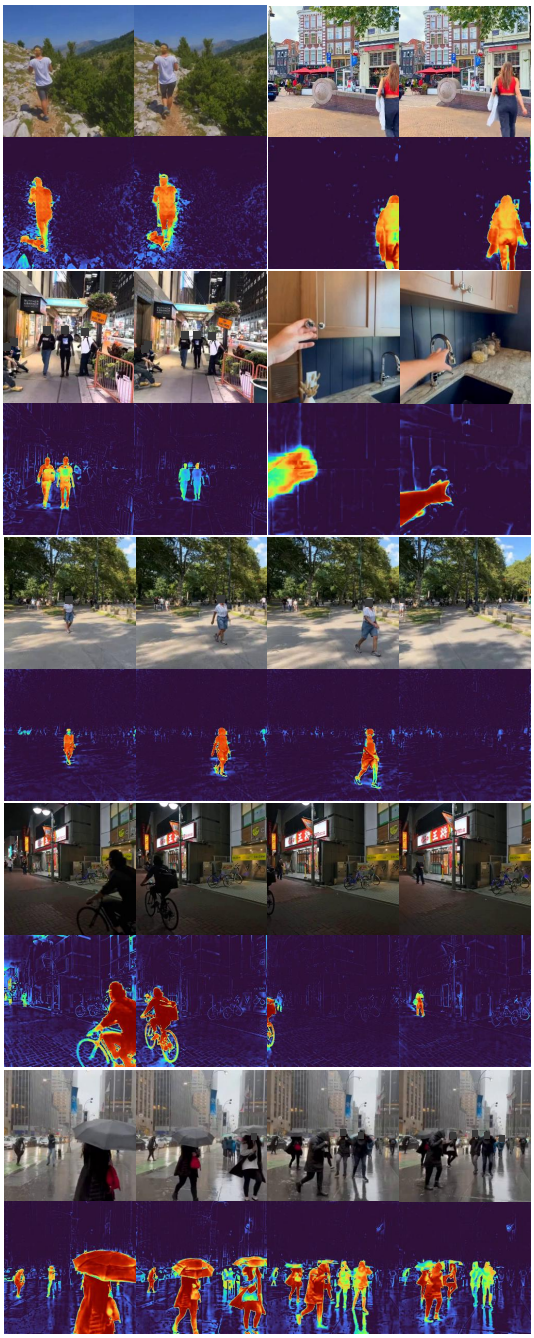


Figure 8: **Self-supervised motion masks.** For each video, the top row shows input frames, and the bottom row shows the raw motion masks M_i predicted by *Scr4tch*, learned without any mask, flow, or motion supervision. Although somewhat noisy, the masks align closely with independently moving content, such as walking people, cyclists, and hands, while static structure remains unmasked despite camera motion.

A.3 Self-Supervised Motion Masks

Figure 8 visualizes the raw motion masks M_i from the mask head on diverse in-the-wild videos. Unlike the probing results in Table 4, these masks involve no labels: they are learned only from image reconstruction and regularization, and are therefore somewhat noisy. Nevertheless, they align closely with independently moving content across walking, cycling, egocentric, and crowded scenes, while static structure remains unmasked as the camera moves. This shows that *Scr4tch* separates scene dynamics from camera motion in a self-supervised manner, providing the basis for learning cameras from static content (Sec. 3.2).

A.4 Ablation Visualizations

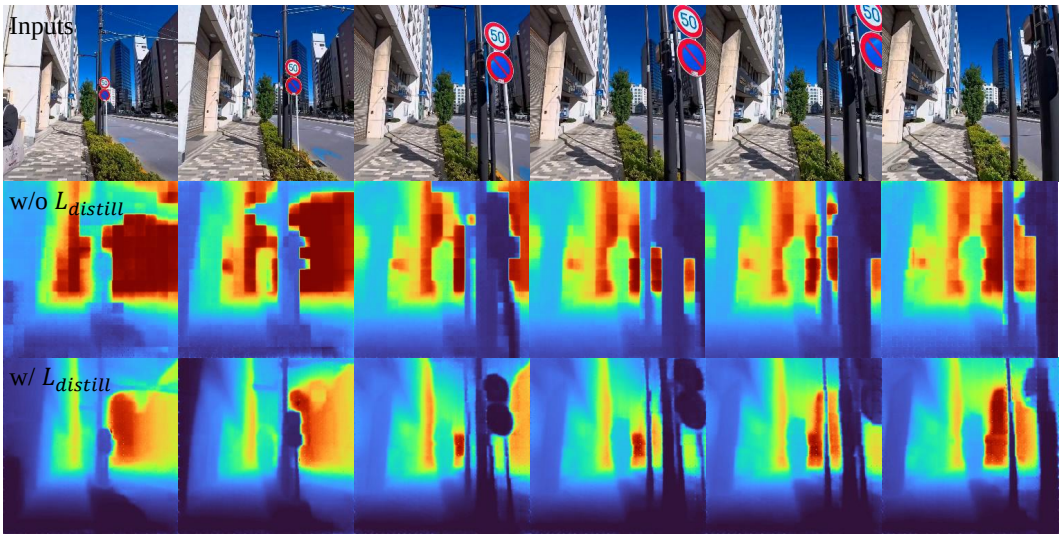


Figure 9: **Effect of rendered-depth distillation on self-supervised depth.** Raw depths D_i predicted directly by the Gaussian head, learned without any depth supervision. Without $\mathcal{L}_{\text{distill}}$ (middle), they retain a patch-grid pattern; with it (bottom), they align with scene content.

Rendered-Depth Distillation. Figure 9 visualizes the raw depths D_i predicted directly by the Gaussian head, which, like all geometry in *Scr4tch*, are learned without depth supervision. These depths receive only indirect gradients through Gaussian splatting, so without $\mathcal{L}_{\text{distill}}$ (Eq. 15) they remain blocky along the patch grid even though the rendered depths converge. Distilling the rendered depths transfers geometric detail back into the shared features, and the raw predictions then follow scene content, including thin structures such as poles and traffic signs.

Camera Gradient Routing. Figure 10 visualizes camera estimation on a Wild-SLAM Mocap sequence in which a cart moves along with the camera and stays nearly fixed in the image. Without CGR, static rendering forces undeformed Gaussians to fit the moving cart, and the resulting camera gradients distort the estimated trajectory. With CGR, static rendering passes camera gradients only through static regions, and the estimated cameras align closely with the ground truth, consistent with the largest translation gain in Table 5.

Single-View and Multi-View Depth Consistency. Figure 11 renders a novel view of a dynamic scene from NVIDIA Dynamic. Without $\mathcal{L}_{\text{sv}}$ (Eq. 16), a wrong depth with compensating motion can still reconstruct the input views of the moving person, but this geometry fails to transfer to other viewpoints: the rendered depth misses the person, and the rendered image shows only faint artifacts at a wrong location. With $\mathcal{L}_{\text{sv}}$, single-view depth constrains the moving person to the correct depth, so the person renders faithfully from the new viewpoint, in line with the largest rendering gain of $\mathcal{L}_{\text{sv}}$ in Table 5.

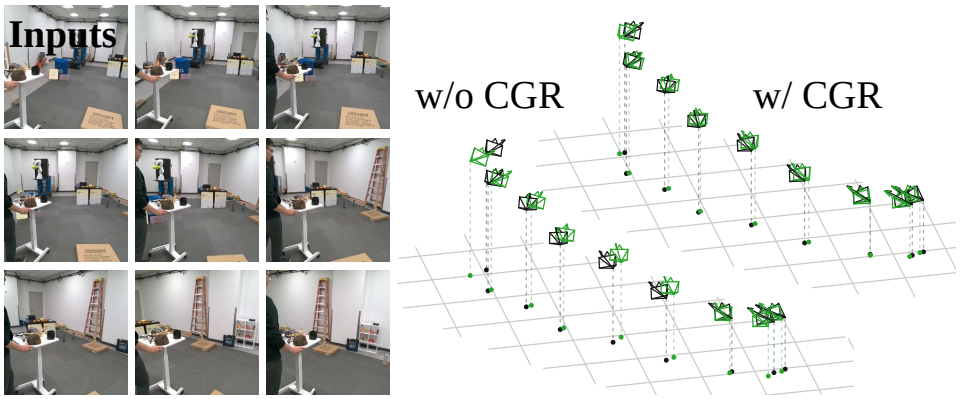


Figure 10: **Effect of camera gradient routing.** Estimated (green) and ground-truth (black) cameras on Wild-SLAM Mocap, where a cart moves along with the camera. Without CGR, the moving cart distorts the estimated cameras; with CGR, they align closely with the ground truth.

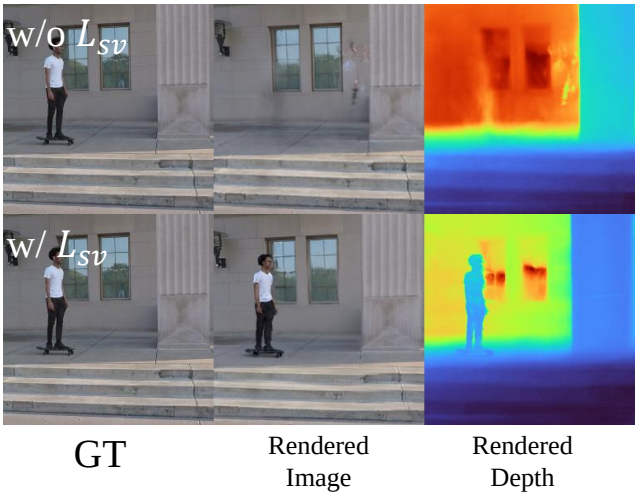


Figure 11: **Effect of single-view and multi-view depth consistency.** A novel view on NVIDIA Dynamic, showing the ground truth, rendered image, and rendered depth. Without $\mathcal{L}_{\text{sv}}$ (top), the moving person takes an incorrect depth and renders only as artifacts at a wrong location; with it (bottom), the person is placed at the correct depth and rendered faithfully.

B Network Architecture

Backbone. We append image-plane coordinates (u, v) to RGB as two additional channels (Liu et al., 2018) and tokenize the resulting images into non-overlapping 16×16 patches. These coordinate channels are the only positional signal; no additional positional embeddings such as RoPE (Su et al., 2021) are used. The transformer consists of 24 pairs of within-view and cross-view self-attention layers, following the alternating-attention design of VGGT (Wang et al., 2025a). To enrich spatial detail without increasing the backbone token count, input tokenization and output refinement each include a fine-resolution branch of four self-attention layers over 8×8-pixel patches of each view, whose features are concatenated with the coarse features along channels.

Camera, Time, and Motion Tokens. Each view carries one camera token and one time token, which join its patch tokens in both within-view and cross-view attention, whereas the $N_{\text{m}} = 64$ global motion tokens participate only in cross-view attention. Camera and time tokens are initialized from embeddings shared across views, and no view-index embeddings or reference-view-specific tokens are used. This preserves equivariance to input-view permutations, similar to the design prin-

ciple of π^3 (Wang et al., 2026c), and means that time tokens are inferred from images rather than assigned from view indices. The time bottleneck projects each time token to three channels, applies LayerNorm, and projects it back to the token dimension. Ignoring the numerical stabilizer of LayerNorm, centering and variance normalization remove two of the three degrees of freedom, leaving one input-dependent degree of freedom.

Transformer Passes. All passes share the same transformer. The bootstrap pass processes the augmented input frames and outputs camera, time, and motion tokens. In latent rendering, target views retain their (u, v) channels with RGB set to zero, and an image head decodes the updated target tokens into RGB. Explicit alignment adds a mask head on the bootstrap features and runs static and dynamic latent rendering, the context-only Gaussian pass (App. C.1), and a single-view pass that applies the Gaussian pass to each input frame independently (App. C.3).

Camera Bottleneck. A camera head h_{cam} decodes each bootstrap camera token into focal lengths f_x and f_y, with the principal point fixed at the image center, and a camera-to-world pose given by a normalized quaternion and a translation vector. A shallow MLP ϕ maps these parameters back into the token space:

$$c_i \xrightarrow{h_{\text{cam}}} \pi_i = (K_i, R_i, t_i) \xrightarrow{\phi} \widetilde{c}_i. \tag{6}$$

Latent pretraining bypasses this bottleneck and retains the full latent camera tokens. The decoded cameras and scene geometry remain subject to monocular scale ambiguity.

C Explicit Alignment Details

We denote context and target view indices by $\mathcal{C}$ and $\mathcal{T}$ and stop-gradient by sg. Pixel-wise losses are averaged over pixels and the relevant views.

C.1 Deformable Gaussian Scene

Context-Only Gaussian Pass. The base Gaussian pass processes only clean context images, without target views, motion tokens, or time tokens. Its camera conditioning is the shared initial camera token, since camera tokens from the bootstrap pass, which observes the targets, could indirectly convey target information. The decoded context cameras are used only for backprojection. Figure 3 repeats a single image three times as input and compares against a separately trained variant that predicts Gaussians in a pass also containing target views. There, jointly estimated camera conditioning can reveal the absence of viewpoint change and allow reconstruction without meaningful depth.

Gaussians and Deformation. For each context pixel p, the Gaussian head predicts a depth parameter $\eta_i(p)$, rotation, scale, opacity, and color. A positive activation ψ gives the depth $D_i(p) = \psi(\eta_i(p))$, and backprojection places the Gaussian center at

$$\mu_i(p) = \mathcal{B}_i(p, D_i(p)), \qquad \mathcal{B}_i(p, d) = R_i\big(dK_i^{-1}\widetilde{p}\big) + t_i, \tag{7}$$

where $\widetilde{p}$ is the homogeneous pixel coordinate. To deform the Gaussians of context view i toward the time of input view j, the deformation head modulates context features F_i^{dyn} from dynamic latent rendering with the updated time token τ_j:

$$\Delta_{i\to j} = g_i \odot h_{\text{def}}\Big(\mathrm{FiLM}(F_i^{\text{dyn}}, \tau_j)\Big), \qquad g_i = 0.1 + 0.8M_i. \tag{8}$$

Its eight output channels update image-plane position, depth, rotation quaternion, and opacity, giving seven degrees of freedom. The deformed center is

$$\mu_{i\to j}(p) = \mathcal{B}_i(p + \Delta p_{i\to j}(p), \psi(\eta_i(p) + \Delta\eta_{i\to j}(p))), \tag{9}$$

and rotation and opacity residuals act before normalization and sigmoid activation, respectively. Dynamic rendering uses the deformed Gaussians to explain the full image. Static rendering uses the undeformed Gaussians with opacity attenuated by $1 - \mathrm{sg}(g_i)$; since this gate is detached, the mask is learned through input mixing, latent blending, deformation gating, and the asymmetric costs.

C.2 Rendering Objectives and Motion Regularization

Latent Objective. The static latent rendering pass receives clean context images and camera conditioning. The dynamic pass additionally receives time tokens and the first ℓ motion tokens, with ℓ sampled uniformly from $\{1, \ldots, N_{\mathrm{m}}\}$ so that every step includes a dynamic rendering alongside the static one. Its context images mix clean and augmented appearance according to the motion mask:

$$I_i^{\mathrm{mix}} = M_i \odot I_i + (1 - M_i) \odot I_i^{\mathrm{aug}}, \qquad i \in \mathcal{C}, \tag{10}$$

where I_i^{aug} is the augmented image used for bootstrapping. The latent objective $\mathcal{L}_{\mathrm{lat}}$ in Eq. 4 uses $\lambda_{\mathcal{A}} = 0.25$.

Explicit Objective and Camera Gradient Routing. For static Gaussian rendering, CGR samples one of two regimes with equal probability: $W = 1$ with both backprojection and rendering cameras detached, or $W = \mathrm{sg}(1 - M)$ with camera gradients enabled. The weighted rendering loss is

$$\mathcal{R}_W(X, I) = W \odot (X - I)^2 + 0.5\ \mathrm{LPIPS}(W \odot X + (1 - W) \odot I, I)\,, \tag{11}$$

which reduces to $\mathcal{R}$ for $W = 1$. Explicit rendering losses are averaged separately over target and context views and weighted by 1 and 0.05, respectively; these view weights are omitted from Eq. 5 for readability.

Motion Regularization. Complementing the asymmetric costs, $\mathcal{L}_{\mathrm{reg}}$ combines L_1 mask sparsity, edge-aware mask smoothness (Ranjan et al., 2019; Lee et al., 2024), and a penalty on deformation in static regions:

$$\mathcal{L}_{\mathrm{reg}} = 0.04\, M_i + 0.04 \sum_{d \in \{x,y\}} |\partial_d M_i|\, e^{-2|\partial_d I_i|} + 0.25\,(1 - g_i) \odot \left|\Delta_{i \to j}^{\mathrm{raw}}\right|, \tag{12}$$

where $\Delta_{i \to j}^{\mathrm{raw}}$ denotes the ungated image-plane and depth residuals.

C.3 Depth Objectives

Dynamic Photometric Reprojection. For each context view i, we deform points backprojected from D_i toward the two temporally nearest other input frames $k \in \mathcal{N}(i)$, which may be contexts or targets, project them with camera π_k, and bilinearly sample RGB:

$$\widehat{I}_{k \to i}(p) = I_k\big(\mathrm{proj}_{\pi_k}\left(\mu_{i \to k}(p)\right)\big)\,. \tag{13}$$

Following the minimum-reprojection principle of Monodepth2 (Godard et al., 2019), we minimize

$$\mathcal{L}_{\mathrm{sample}} = \min_{k \in \mathcal{N}(i)} \mathrm{MSE}_{\mathrm{RGB}}\Big(\widehat{I}_{k \to i}(p), I_i(p)\Big)\,, \qquad i \in \mathcal{C}, \tag{14}$$

masking pixels without a valid projection. With two input frames, the single other frame serves as the reference.

Rendered-Depth Distillation. We distill the context depth D_i^{render} rendered from the deformed Gaussians into D_i:

$$\mathcal{L}_{\mathrm{distill}} = \rho\big(D_i, \mathrm{sg}(D_i^{\mathrm{render}})\big)\,, \quad i \in \mathcal{C}, \qquad \rho(x, y) = \mathrm{smooth}_{L_1}(\log x - \log y). \tag{15}$$

Single-View and Multi-View Depth Consistency. The single-view pass uses only the shared initial camera token and no motion or time tokens, taking clean or augmented frames with equal probability. Its predictions D_i^{single} are coupled to the rendered depths D_i^{render} at every input view through

$$\mathcal{L}_{\mathrm{sv}} = m_i \odot \rho\Big(D_i^{\mathrm{render}}, \mathrm{sg}(D_i^{\mathrm{single}})\Big) + (1 - m_i) \odot \rho\Big(D_i^{\mathrm{single}}, \mathrm{sg}(D_i^{\mathrm{render}})\Big)\,, \tag{16}$$

where $m_i = \mathrm{sg}(M_i)$, so that single-view depth supervises moving regions and multi-view rendered depth supervises static regions.

C.4 Combined Objective

Explicit alignment jointly optimizes the shared transformer and all prediction heads with

$$\mathcal{L}_{\text{align}} = \mathcal{L}_{\text{lat}} + \mathcal{L}_{\text{exp}} + \mathcal{L}_{\text{sample}} + \mathcal{L}_{\text{distill}} + \mathcal{L}_{\text{sv}} + \mathcal{L}_{\text{reg}}, \tag{17}$$

where $\mathcal{L}_{\text{lat}}$ and $\mathcal{L}_{\text{exp}}$ are given in Eqs. 4 and 5 and $\mathcal{L}_{\text{reg}}$ in Eq. 12. Latent pretraining minimizes only $\mathcal{L}_{\text{dyn}}^{\text{lat}}$ (Sec. 3.1). Neither stage uses camera annotations, offline geometric pseudo-labels, or pretrained geometric models.

D Implementation Details

Training. Frames are sampled with random temporal strides. To partition a frame, we divide it into four quadrants in random order and form each split from two of them. Bootstrap augmentation applies, with equal probability, either blurring, down-up sampling, color jitter, and partial grayscale conversion, or grayscale conversion with random channel weights, contrast, polarity, and smooth local intensity offsets. Both stages use AdamW (Loshchilov & Hutter, 2019) with $\beta = (0.9, 0.95)$, linear warmup, cosine decay, and bf16 mixed precision for 100k iterations, on 8 GPUs with 4-step gradient accumulation for about three days each. Latent pretraining uses an effective batch size of 512 and a learning rate decaying from 8×10^{-4} to 1×10^{-4}. Explicit alignment is initialized from latent pretraining, keeps nested dropout, cross-split exchange, and appearance augmentation, and uses an effective batch size of 256 and a learning rate decaying from 2×10^{-4} to 2×10^{-5}. At inference, input frames are not augmented, all motion tokens are used, and camera estimation and static NVS bootstrap from full images.

Camera Estimation. On Wild-SLAM Mocap, we subsample every 24th frame and run *Scr4tch* on 9-frame windows that share their endpoints, so that every adjacent pair lies within one window. On DL3DV-Evaluation, we use 53 scenes and take every sixth frame as a context, evaluating all pairs of adjacent contexts. RRA@τ and RTA@τ count pairs whose relative rotation error and translation direction error fall below τ degrees, ignoring the sign of the translation direction, without Sim(3) alignment.

Novel View Synthesis. On DL3DV-Evaluation, each target lies midway between two adjacent contexts. We align the predicted context cameras to their ground truth with a similarity transform and map the ground-truth target poses into the predicted frame; rendering uses the mean predicted context intrinsics. Ours$^{\text{lat}}$ conditions latent rendering on the remapped tokens $\phi(\pi_j)$ of the aligned target cameras, and Ours$^{\text{exp}}$ renders the Gaussians predicted from the context images.

Probing. Depth and motion segmentation probing share the same head: two convolutional residual blocks on a 32-channel feature map at input resolution. For every method, the frozen backbone features are mapped to this shape by a learnable linear adapter; when multiple features are used, each receives an equal share of the 32 channels, and the results are concatenated.

E Casual YouTube



Figure 12: **Examples from *Casual YouTube*.** Casually captured monocular videos under diverse scenes, weather, and lighting, used as raw footage without geometric preprocessing.

Figure 12 shows examples from *Casual YouTube*, our 2,154-hour collection of casually captured monocular videos. It covers walking, cycling, and first-person driving under sunny, rainy, snowy,

and nighttime conditions, often with crowds, snowfall, and low light. The videos are used as raw footage, without camera calibration, geometric annotations, or any other geometric preprocessing. Despite these challenges, adding *Casual YouTube* to training improves both camera estimation and dynamic NVS (Table 5), suggesting that *Scr4tch* can scale with unprocessed Internet video.