TA-GS:Transient-Aware Gaussian Splatting for Robust Static 3D Reconstruction

Peng Chen Jingyuan Xu Mingrui Li Hongyu Wang*
Dalian University of Technology
Dalian, China
{32409051, mmclmr, 2905450254}@mail.dlut.edu.cn, whyu@dlut.edu.cn

Abstract

Reconstructing static 3D scenes from image sequences affected by transient interference remains a fundamental challenge in computer vision. Existing methods for transient interference suppression typically operate at the reconstruction or rendering stage, overlooking the fact that the traditional SfM process is already affected by such interference, which introduces errors in camera poses and geometric structure, ultimately degrading overall reconstruction quality. In addition, current approaches for generating transient masks rely on limited information, primarily pixel-level and semantic features, resulting in inaccurate transient region identification. To address these limitations, we propose Transient-Aware Gaussian Splatting (TA-GS), a robust static scene reconstruction framework based on 3D Gaussian Splatting for transient environments. TA-GS introduces a geometry-first probabilistic SfM built upon MASt3R priors, establishing a stable and accurate geometric foundation from the source. We further design a transient-aware mask learning module that integrates optical flow residuals and depth cues, supervised by DINOv2 semantic differences, to achieve precise pixel-level transient detection. Finally, static probabilities and transient masks jointly guide the initialization, pruning, and rendering optimization of 3D Gaussian Splatting. Experiments demonstrate that TA-GS significantly improves the stability, transient region identification accuracy, and fidelity of static scene reconstruction in diverse and challenging transient scenarios.

Keywords: 3D Reconstruction, 3D Gaussian Splatting, Robustness, Transient Interference

1. Introduction

Reconstructing high-fidelity 3D scenes from a limited set of multi-view images has long been a central goal in the field of computer vision and has significant applications in domains such as autonomous driving, augmented reality, virtual reality, robotic navigation, and 3D map construction. Specifically, the task involves generating a multi-view consistent 3D scene representation from a set of input images with known camera poses, thereby enabling photo-realistic rendering from novel viewpoints. In recent years, representative methods have emerged, significantly advancing the state of 3D reconstruction techniques.

(a) Original (b) Ours

Figure 1: Visual comparison of reconstruction results for scenes with transient disturbances. Our method effectively suppresses the influence of transient elements, producing complete and consistent static scene reconstructions with high-quality detail recovery.

NeRF [15], a breakthrough method in 3D reconstruction, innovatively models the radiance field of a scene implicitly using neural networks (MLPs) to generate high-quality images. Unlike traditional explicit geometric representations, NeRF leverages deep learning to nonlinearly map lighting and color information, enabling realistic rendering from multiple viewpoints. Its core strength lies in volume rendering, which accurately captures subtle lighting variations, reflections, and refractions, thereby enhancing image realism and detail. However, NeRF suffers from slow rendering speed.Subsequent research [6, 16, 11] has proposed various

*Corresponding author

improvements to enhance its efficiency and performance.

In contrast, 3D Gaussian Splatting (3DGS)[8] employs an explicit representation based on Gaussian distributions, representing the 3D scene as a set of Gaussian points with spatial positions and color attributes. Each Gaussian point corresponds to a local region in the scene and reconstructs the overall scene by integrating information from multi-view images. Compared with NeRF's implicit representation, this method offers higher computational efficiency, making it suitable for real-time rendering and transient scene processing. It also handles large datasets more effectively and provides rapid rendering. Furthermore, the explicit point cloud representation enhances robustness and scalability in large-scale scene reconstruction, making it suitable for complex 3D reconstruction tasks.

Various extensions of 3DGS [5, 7, 28] have been developed to improve efficiency, scalability, and geometric fidelity. Notably, GSplat [27] provides a highly optimized differentiable rasterization backend that significantly accelerates training and rendering speeds. These methods collectively enhance 3DGS in terms of rendering performance, geometric detail, and adaptability to large-scale real-world scenes.

In real-world environments, capturing ideal image sequences is often infeasible, particularly in scenes affected by transient interference, where moving objects, lighting fluctuations, or occlusions frequently occur. These transient factors can lead to inconsistent correspondences, unstable depth estimation, and mismatched geometry, ultimately resulting in blurring, ghosting, or structural distortions in the reconstructed scenes.

In the field of transient noise removal for 3D reconstruction, existing methods based on 3DGS [13, 4, 9, 20] have achieved significant progress in improving reconstruction accuracy and detail fidelity. However, they still exhibit several limitations. First, these methods typically do not sufficiently consider the impact of transient point noise during the Structure-from-Motion (SfM) or multi-view geometry initialization stages. Because transient noise points are mistakenly incorporated into the initial sparse point cloud, errors may accumulate throughout the reconstruction process, resulting in deviations in the final scene geometry—especially in transient or semi-transient scenarios. Second, existing approaches for mask prediction mainly rely on a single source of information, such as pixel intensity or semantic priors alone. This single-source dependence often leads to inaccurate mask predictions, manifested as either under-masking or over-masking, which further affects the removal of transient noise and the overall quality of reconstruction.

To address this, we propose an innovative 3D reconstruction method, Transient-Aware 3D Gaussian Splatting (TA-GS), which effectively recovers high-quality 3D scenes from image sequences affected by transient interference. Based on MASt3R[10], our method designs a probabilistic deep-learning SfM model that jointly estimates camera poses, 3D structure, and per-point static probabilities, inherently mitigating the impact of transient elements from a geometric perspective. To balance efficiency and robustness on large-scale image collections, we adopt the sparse graph construction strategy from MASt3R-SfM [2] to build a sparse yet stable scene graph. In this structure, keyframes form a fully connected backbone to ensure global consistency and local accuracy. Furthermore, we introduce a transient-aware mask predictor that leverages motion-depth features combined with DINOv2-based semantic consistency to identify view-dependent transient regions, complementing the static probabilities obtained from geometry. Finally, we integrate these two sources of information into a transient-aware 3D Gaussian point rendering pipeline, explicitly modeling transient suppression during initialization, pruning, and loss computation, thereby effectively eliminating transient artifacts while preserving high-fidelity static reconstruction.

2. Related work

2.1. The Revolution in SfM

Traditional SfM pipelines rely on a sequential process of local feature detection, matching, geometric verification (e.g., via RANSAC [3]), incremental triangulation, and bundle adjustment [23, 25]. While highly successful, this pipeline is inherently fragile; errors in feature matching or outlier rejection can propagate, leading to complete failure in challenging conditions such as textureless surfaces, repetitive patterns, or significant occlusions [22].

The field was revolutionized by the introduction of DUSt3R[24], which framed multi-view reconstruction as a direct, dense regression problem. By leveraging transformer-based architecture, DUSt3R directly predicts accurate depth and camera pose for every pixel in an unconstrained setting, eliminating the need for explicit feature matching and robust estimation. This end-to-end approach demonstrated remarkable robustness and speed but was primarily focused on geometric reconstruction.

Building upon this foundation, MASt3R[10] extends the DUSt3R framework by integrating a powerful dense feature matching head trained with contrastive losses. This critical advancement enables the model to not only perform dense 3D reconstruction but also produce highly-accurate and robust local features. By unifying precise geometry and reliable matching into a single, cohesive model, MASt3R effectively bridges the gap between the robustness of data-driven 3D reconstruction and the precision required for tasks like high-quality SfM.

2.2. Robustness in NeRF

Several methods have been proposed to improve the robustness of NeRF under transient objects and photometric disturbances. NeRF-W [14] decomposes each training image into a static radiance field and a per-image transient component, effectively filtering pedestrians, vehicles, and illumination variations in Internet photo collections. NeRF-on-the-go [19] adapts NeRF for casual mobile capture, jointly learning transient segmentation and static scene reconstruction to handle frequent occlusions in handheld video. Robust optimization methods such as RobustNeRF [21] treat inconsistent pixels as outliers and adopt robust loss functions to reduce their influence on reconstruction. RegNeRF [17] applies regularization to improve generalization under sparse views, implicitly suppressing the effect of noisy observations. These techniques together enhance NeRF's ability to reconstruct clean geometry and appearance in challenging real-world scenarios.

2.3. Robustness in 3DGS

Several extensions of 3D Gaussian Splatting (3DGS) have been proposed to handle transient objects, transient occlusions, and appearance changes. T-3DGS [13] introduces an unsupervised uncertainty predictor together with a SAM2-based mask optimizer, enabling accurate detection and removal of transient objects without additional priors for high-quality static reconstruction. RobustSplat [4] delays Gaussian growth and employs a multi-scale mask-guided strategy to prioritize static structure optimization, effectively reducing artifacts caused by moving distractors. HybridGS [12] combines 2D and 3D Gaussian representations to disentangle transient and static components, significantly improving reconstruction quality under complex transient environments. ForestSplats [18] leverages a deformable transient field and superpixel-aware masks to precisely model and separate transient occluders for robust scene recovery. WildGaussians [9] introduces per-Gaussian and per-image learnable appearance embeddings and a lightweight MLP for color transformation, while using DINOv2 feature cosine similarity for robust uncertainty prediction, allowing 3DGS to handle challenging outdoor videos with varying appearances and transient occlusions while maintaining real-time rendering speed. SpotLessSplats [20] learns transient anomaly masks from features of a pre-trained semantic model via an MLP, guiding optimization to focus on clean static backgrounds. These techniques collectively enhance the robustness of 3DGS in the presence of transient interference, enabling high-fidelity and artifact-free static scene reconstruction in real-world scenarios.

3. Method

3.1. Preliminary

3.1.1 3D Gaussian splatting

3D Gaussian Splatting[8]is an efficient approach to model 3D scenes by representing geometry as a collection of 3D Gaussian primitives, thus avoiding explicit surface normals. This representation enables photorealistic novel view synthesis. Each Gaussian is described by its center $\mu_i \in \mathbb{R}^3$, covariance matrix $\Sigma_i \in \mathbb{R}^{3\times3}$, opacity parameter $\alpha_i \in \mathbb{R}$, and color coefficient $x_i \in \mathbb{R}^3$. The density of a single Gaussian is defined as:

$$G(z) = \exp\Big(-\frac{1}{2}z^\top \Sigma_i^{-1} z\Big). \tag{1}$$

In differentiable rasterization, the Gaussian is weighted by α_i to determine its image-space contribution. To project the Gaussians into the camera plane, the covariance is expressed in camera coordinates using the world-to-camera transform W and the projection Jacobian J:

$$\Sigma_i' = JW\Sigma_i W^\top J^\top. \tag{2}$$

To ensure Σ_i remains positive semi-definite and supports anisotropic scaling, we decompose it as:

$$\Sigma_i = R_i S_i S_i^\top R_i^\top, \tag{3}$$

where R_i is a rotation matrix and S_i is a learnable scale matrix.

Finally, the pixel color C is computed by front-to-back alpha compositing over all Gaussians $i \in \mathcal{N}$ that project to the pixel:

$$C = \sum_{i\in\mathcal{N}} x_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j), \tag{4}$$

where $\mathcal{N}$ is sorted by depth. This accumulation scheme ensures proper occlusion handling and smooth blending.

3.1.2 MASt3R

MASt3R [10] is a 3D-grounded image matching approach built upon the DUSt3R framework. The core preliminary is the joint problem of 3D reconstruction and camera calibration from uncalibrated image pairs. Given two images I^1 and I^2, the network regresses two dense *pointmaps* $X^{1,1}$ and $X^{2,1} \in \mathbb{R}^{H\times W\times3}$. A pointmap $X^{a,b}$ maps each pixel in image I^a to its corresponding 3D point, expressed in the coordinate system of camera $\mathcal{C}^b$. By regressing both pointmaps into a shared coordinate system (e.g., that of $\mathcal{C}^1$), the framework implicitly solves for relative pose and scene geometry.

This regression is supervised by a scale-invariant regression loss. For a view v and pixel i with ground-truth 3D point $\hat{X}_i^{v,1}$, the loss term is:

$$\ell_{\text{regr}}(v,i) = \left\| \frac{1}{z} X_i^{v,1} - \frac{1}{\hat{z}} \hat{X}_i^{v,1} \right\| \tag{5}$$

where z and $\hat{z}$ are scale normalization factors (typically the mean depth of all valid points). The final objective $\mathcal{L}_{\text{conf}}$ is a confidence-weighted version of this loss, where the network also learns a per-pixel confidence C_i^ν:

$$\mathcal{L}_{\text{conf}} = \sum_{v\in\{1,2\}} \sum_{i\in\mathcal{V}^v} [C_i^\nu \ell_{\text{regr}}(v,i) - \alpha \log C_i^\nu] \tag{6}$$

MASt3R augments this 3D foundation by adding a new head to predict dense local features, trained with an additional contrastive matching loss, thereby combining the robustness of 3D geometric reasoning with the precision of learned feature matching.

3.2. Overview

We propose a novel framework for reconstructing static 3D scenes from image collections affected by transient interference. The overall pipeline is illustrated in Fig. 2. Transient points in a scene can undermine the accuracy of 3D reconstruction. To address this, we develop *Geometry-First Probabilistic SfM*. Since relying on a single visual cue is insufficient to accurately capture transient or uncertain regions, we introduce rich optical flow and depth information, combined with semantic supervision, to propose *Transient-Aware Mask Prediction*, aiming to generate more accurate pixel-level masks. Finally, transient artifacts limit the fidelity of the reconstructed model, which is mitigated by *Refined 3D Gaussian Splatting Reconstruction*. The framework consists of three main components:

1. **Geometry-First Probabilistic SfM**: Using Expectation Maximization, camera poses, 3D structure, and point-wise static probabilities are jointly estimated, providing a robust geometric foundation for reconstruction;

2. **Transient-Aware Mask Prediction**: Pixel-level masks are predicted by integrating motion-depth cues and supervised using semantic consistency extracted from DINOv2 features, guiding the identification of transient or uncertain regions;

3. **Refined 3D Gaussian Splatting Reconstruction**: Both static probabilities and learned masks are leveraged to produce high-fidelity 3D reconstructions while suppressing transient artifacts.

3.3. Geometry-First Probabilistic SfM

We present a geometry-first probabilistic Structure-from-Motion (SfM) framework for robust 3D scene reconstruction in the presence of transient objects. Built upon a sparse scene graph of carefully selected image pairs, our method jointly estimates camera poses, 3D point locations, and static/transient labels. The proposed framework, detailed in our probabilistic generative model (Sec.3.3.1) and EM-based MAP inference procedure (Sec.3.3.2), achieves robustness by seamlessly unifying geometric reprojection constraints with prior knowledge about point stability, thereby effectively suppressing transient outliers.The overall EM-based inference procedure of our geometry-first probabilistic SfM framework is summarized in Algorithm 1.

3.3.1 Probabilistic Generative Model Formulation

Let $\mathcal{V} = \{I^n\}_{n=1}^N$ represent an unordered collection of N input images. Our goal is to recover the parameters of static cameras $\mathcal{P} = \{P_n\}$, a set of 3D points $\mathcal{X} = \{\mathbf{X}_a\}$ representing the scene structure, and, crucially, a set of binary latent variables $\mathcal{S} = \{s_a\}$ where $s_a = 1$ indicates that point a is static.

Our encoder and decoder architectures are directly inherited from the MASt3R prior [10], forming an Encoder–Graph–Decoder architecture, where the "Graph" refers to a sparse scene graph connecting images based on visual overlap, upon which we introduce a probabilistic modeling layer.

Each image is first processed by a shared visual encoder f_{enc} to obtain dense feature maps $\mathcal{F} = \{F^n = f_{\text{enc}}(I^n)\}_{n=1}^N$, where $F^n \in \mathbb{R}^{H\times W\times D}$ denotes the latent representation of image n. These features are used both for image retrieval and for dense correspondence estimation.

The sparse scene graph $\mathcal{E}$ is then constructed from the feature collection $\mathcal{F}$ using a scalable strategy inspired by MASt3R-SfM [2]. Candidate image pairs are retrieved via ASMK similarity, farthest-point-sampled keyframes form a fully connected clique to establish a stable global backbone, and additional k-nearest-neighbor edges maintain local connectivity. This design ensures a globally consistent yet computationally efficient graph structure.

The decoder then estimates the initial camera poses $\mathcal{P} = \{P_n\}$ and a dense 3D structure $\mathcal{X} = \{\mathbf{X}_a\}$ from these image correspondences. Building upon this deterministic reconstruction, we introduce a probabilistic layer that models uncertainty in both scene structure and point staticness. Specifically, each 3D point $\mathbf{X}_a$ is associated with a binary latent variable $s_a \in \{0,1\}$, indicating whether it is static or transient. This extends the MASt3R decoder into a generative probabilistic model that reasons jointly about geometry and motion consistency.

For each connected image pair $(n,m) \in \mathcal{E}$, the MASt3R

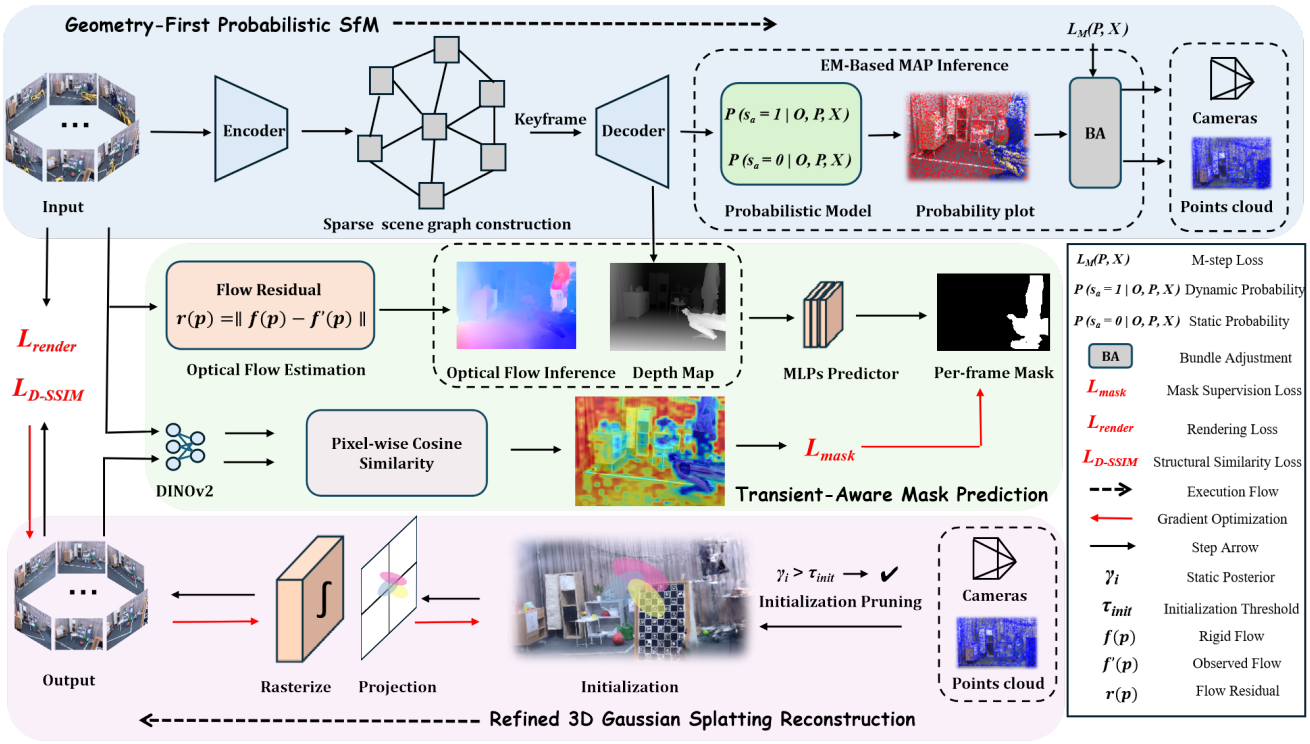

Figure 2: Overview of the proposed pipeline for static 3D scene reconstruction in the presence of transient interference. The diagram is arranged in a vertical structure: the top component, *Geometry-First Probabilistic SfM*, first estimates camera poses, 3D structure, and point-wise static probabilities; the middle component, *Transient-Aware Mask Prediction*, identifies pixel-level transient or uncertain regions; and the bottom component, *Refined 3D Gaussian Splatting Reconstruction*, leverages this information to produce high-fidelity static 3D reconstructions while suppressing transient artifacts.

matching head produces a set of feature correspondences

$$\mathcal{M}^{n,m} = \{o_c = (\mathbf{y}_c^n, \mathbf{y}_c^m, q_c)\}, \tag{7}$$

where $(\mathbf{y}_c^n, \mathbf{y}_c^m)$ are pixel coordinates and q_c is the MASt3R confidence score. Collectively, these correspondences constitute the observed data

$$\mathbf{O} = \{\mathcal{M}^{n,m} \mid (n,m) \in \mathcal{E}\}. \tag{8}$$

The joint distribution over all latent variables and observations factors as

$$P(\Theta, \mathbf{O}) = P(\mathcal{P})P(\mathcal{X})\,P(\mathcal{S})\,P(\mathbf{O} \mid \mathcal{P}, \mathcal{X}, \mathcal{S}), \tag{9}$$

where uninformative priors $P(\mathcal{P})$ and $P(\mathcal{X})$ are assumed for camera poses and 3D structure.

The prior over the staticness labels $\mathcal{S}$ follows a Bernoulli distribution with a globally shared parameter ρ, reflecting the prior belief that most points in a typical scene are static:

$$s_a \sim \text{Bernoulli}(\rho), \tag{10}$$

or equivalently,

$$P(s_a) = \rho^{s_a}(1-\rho)^{1-s_a}, \quad s_a \in \{0,1\}. \tag{11}$$

The probabilistic model defines the likelihood of each observed match conditioned on $\mathcal{P}$, $\mathcal{X}$, and $\mathcal{S}$. For static points ($s_a = 1$), the observation is expected to satisfy geometric consistency across views, leading to a Gaussian reprojection likelihood weighted by the confidence q_c:

$$P(o_c \mid s_a = 1, \mathcal{P}, \mathcal{X}) \propto \exp\left(-\frac{q_c}{2\sigma_s^2}\|\pi_n(\mathbf{X}_a) - \mathbf{y}_c^n\|_2^2\right), \tag{12}$$

where $\pi_n(\cdot)$ denotes the projection function for camera P_n. For transient points ($s_a = 0$), correspondences are assumed to follow a uniform distribution over the image domain Ω:

$$P(o_c \mid s_a = 0, \mathcal{P}, \mathcal{X}) = \mathcal{U}(\Omega) = \frac{1}{|\Omega|}. \tag{13}$$

This formulation captures a key geometric intuition: static points exhibit small reprojection errors when camera parameters are consistent, whereas transient points have no coherent multi-view geometry and are therefore modeled as uniformly distributed noise. Together, these components define a fully probabilistic structure-from-motion model that augments the MASt3R geometric decoder with uncertainty-aware reasoning over static and transient scene elements.

Algorithm 1 Geometry-First Probabilistic SfM

Require: Images $\mathcal{V} = \{I^n\}_{n=1}^N$, prior ρ, variance σ_s^2
Ensure: Camera poses $\mathcal{P}$, 3D points $\mathcal{X}$, static probs $\{\gamma_a\}$

Feature Encoding:
for each image $I^n \in \mathcal{V}$ **do**
  $F^n \leftarrow f_{\text{enc}}(I^n)$
end for
Sparse Scene Graph:
Build edges $\mathcal{E}$ via ASMK retrieval, keyframe clique, and k-NN connectivity.
Initialization:
Extract correspondences $\mathcal{M}^{n,m}$ for $(n,m) \in \mathcal{E}$ using MASt3R;
Estimate initial $(\mathcal{P}^{(0)}, \mathcal{X}^{(0)})$ from MASt3R decoder;
Initialize $\gamma_a \leftarrow \rho$.
for $t = 1$ to T **do**
  E-step: compute static posterior for each point a
  $L_{\text{static}}(a) = \sum_c \frac{q_c}{2\sigma_s^2} \|\pi_n(\mathbf{X}_a) - \mathbf{y}_c^n\|_2^2$
  $\gamma_a = \frac{\rho e^{-L_{\text{static}}(a)}}{\rho e^{-L_{\text{static}}(a)} + (1-\rho)C}$
  M-step: weighted bundle adjustment
  Solve $\min_{\mathcal{P},\mathcal{X}} \sum_{(a,c)} w_{a,c} \|\pi_n(\mathbf{X}_a) - \mathbf{y}_c^n\|_2^2$
  where $w_{a,c} = \frac{\gamma_a q_c}{2\sigma_s^2}$, using sparse LM (e.g., Ceres).
end for
return $\mathcal{P}, \mathcal{X}, \gamma_a$

3.3.2 EM-Based MAP Inference

Based on the probabilistic generative model formulated above, we employ the Expectation-Maximization (EM) algorithm to compute Maximum a Posteriori (MAP) estimates of the latent variables.

Specifically, the EM algorithm iteratively infers the static probabilities of points (**E-step**) and refines camera poses and 3D structure (**M-step**) under the probabilistic assumptions defined in the generative model. This establishes a robust geometric foundation that mitigates the influence of transient points, providing reliable priors for subsequent transient-aware mask prediction and 3D Gaussian Splatting reconstruction.

E-Step: Given the current estimates of camera parameters $\mathcal{P} = \{P_n\}$ and 3D structure $\mathcal{X} = \{\mathbf{X}_a\}$, we compute the posterior probability γ_a that point a is static: $\gamma_a = P(s_a = 1 \mid \mathbf{O}, \mathcal{P}, \mathcal{X})$.

This posterior probability is computed as

$$\gamma_a = \frac{\rho \, \exp\big(-L_{\text{static}}(a; \mathcal{P}, \mathcal{X})\big)}{\rho \, \exp\big(-L_{\text{static}}(a; \mathcal{P}, \mathcal{X})\big) + (1-\rho)\, C}, \tag{14}$$

where the cumulative weighted reprojection error of point a is

$$L_{\text{static}}(a; \mathcal{P}, \mathcal{X}) = \sum_c \frac{q_c}{2\sigma_s^2} \big\|\pi_n(\mathbf{X}_a; P_n) - \mathbf{y}_c^n\big\|_2^2. \tag{15}$$

Here, C is a constant arising from the uniform likelihood for transient points, and γ_a can be interpreted as the probability that point a is static given the current estimates of cameras and 3D structure.

M-Step: Holding γ_a fixed, we update camera parameters and 3D points by minimizing the expected negative log-posterior:

$$\mathcal{L}_{\text{M}}(\mathcal{P}, \mathcal{X}) = \sum_a \gamma_a \cdot L_{\text{static}}(a; \mathcal{P}, \mathcal{X}) + \lambda \sum_a (1 - \gamma_a), \tag{16}$$

where $\lambda = -\log(\rho/(1-\rho))$. Since the second term does not depend on $(\mathcal{P}, \mathcal{X})$, the optimization effectively reduces to

$$\min_{\mathcal{P},\mathcal{X}} \sum_{(a,c)} w_{a,c} \big\|\pi_n(\mathbf{X}_a; P_n) - \mathbf{y}_c^n\big\|_2^2, \qquad w_{a,c} = \frac{\gamma_a q_c}{2\sigma_s^2}, \tag{17}$$

which is equivalent to a weighted bundle adjustment (BA) problem. Each residual is scaled by the posterior probability γ_a, so points likely to be transient ($\gamma_a \approx 0$) exert negligible influence, while highly static points dominate the refinement of camera poses and 3D structure.

In practice, we solve this weighted BA using the standard sparse Levenberg–Marquardt algorithm with Schur complement elimination, as implemented in Ceres Solver [1]. This formulation ensures efficient EM updates, progressively improving camera accuracy while suppressing transient outliers.

Distinction from Standard Outlier Rejection:It is important to distinguish our approach from traditional robust SfM methods that rely on RANSAC-based 'hard' filtering. In such pipelines, outlier rejection is a binary and irreversible decision: points classified as outliers are discarded before reconstruction begins, potentially leading to the loss of valid fine-grained structures that were mistakenly rejected due to noise.

In contrast, our probabilistic formulation retains the uncertainty information for every point in the form of a continuous posterior γ_a. Crucially, this soft probability is not merely used for filtering but is explicitly integrated into the subsequent 3DGS optimization as a survival weight. This design creates a differentiable feedback loop where points that are geometrically ambiguous in the SfM stage can still survive and 'solidify' if they contribute positively to the photometric rendering loss. This capability to recover false negatives from the initialization stage represents a fundamental advantage over standard hard-filtering techniques.

3.4. Transient-Aware Mask Prediction

The probabilistic SfM stage provides point-wise static probabilities γ_a, which capture long-term geometric consistency across views. However, these probabilities are defined in 3D space and cannot directly model localized,

view-specific inconsistencies. For rendering-level suppression of dynamics, we require per-pixel masks that adapt to each image. To this end, we propose a transient-aware mask learning module that fuses motion and geometry cues with semantic-level supervision.

To accurately predict transient pixels in transient scenes, we integrate motion-depth feature extraction with a semantic consistency supervision mechanism. By jointly leveraging motion cues and depth information, our approach generates pixel-wise transient masks(Sec.3.4.1), while simultaneously utilizing semantic features from DINOv2 to provide robust supervisory signals for mask prediction(Sec.3.4.2).The overall motion–depth fusion and semantic consistency learning process is summarized in Algorithm 2.

3.4.1 Transient Mask Prediction via Motion-Depth Feature Fusion

To disentangle true object motion from camera-induced displacement, we leverage per-frame depth maps D^n generated by our geometric decoder alongside corresponding camera poses. Given two frames I^n and I^m with relative pose $(R_{n,m}, t_{n,m})$, a pixel $\mathbf{p}$ in I^n is projected into I^m as

$$\hat{\mathbf{p}}' = \pi \left(R_{n,m} \pi^{-1}(\mathbf{p}, D^n(\mathbf{p})) + t_{n,m} \right), \tag{18}$$

yielding the rigid flow

$$\hat{\mathbf{f}}(\mathbf{p}) = \hat{\mathbf{p}}' - \mathbf{p}. \tag{19}$$

Meanwhile, we estimate the observed optical flow $\mathbf{f}(\mathbf{p})$ using GMFlow [26]. The residual

$$r(\mathbf{p}) = \|\mathbf{f}(\mathbf{p}) - \hat{\mathbf{f}}(\mathbf{p})\|_2 \tag{20}$$

quantifies how much the pixel motion deviates from rigid scene geometry: small values indicate static regions, while large values suggest transient objects.

Building on this, we construct a motion–depth descriptor that integrates complementary cues at each pixel:

(i) **Rigid-flow residual** $r(\mathbf{p})$: captures the discrepancy between estimated rigid flow and observed optical flow, aiding static-transient pixel discrimination.

(ii) **Magnitude and orientation of observed flow** $\mathbf{f}(\mathbf{p})$: encodes local motion patterns.

(iii) **Normalized depth** $\tilde{D}^n(\mathbf{p})$: provides scene-scale context.

(iv) **Depth gradient** $\nabla D^n(\mathbf{p}) = (\partial D^n/\partial x, \partial D^n/\partial y)$: computed directly from decoder outputs to emphasize structural boundaries.

Concatenating these channels yields a compact representation

$$F_{\text{dyn}}(\mathbf{p}) \in \mathbb{R}^C, \tag{21}$$

which is processed by a lightweight predictor f_θ to produce a per-pixel transient probability:

$$m(\mathbf{p}) = f_\theta(F_{\text{dyn}}(\mathbf{p})). \tag{22}$$

This design enables the predictor to jointly reason about geometric consistency, local motion, and structural context, allowing robust identification of transient pixels even under challenging camera motion. Notably, all depth-related features are derived from the decoder, ensuring consistency with the reconstructed 3D scene. Specifically, while the rigid flow and depth gradient are computed from the raw depth D^n, the normalized depth $\tilde{D}^n$ is obtained by min-max scaling D^n to a unit range. This normalization balances the transient range across feature channels and promotes stable learning, without altering the structural information vital for motion-depth fusion.

3.4.2 Semantic Consistency Supervision via DINOv2

Low-level flow–depth residuals may be noisy due to depth errors, illumination changes, or optical flow drift. To provide a robust supervisory signal, we leverage semantic features from DINOv2. For each image I^n and its corresponding rendered image $\hat{I}^n$, we extract feature maps $\Phi(I^n), \Phi(\hat{I}^n)$ with a frozen DINOv2 backbone. The cosine similarity at pixel $\mathbf{p}$ is computed as:

$$s(\mathbf{p}) = \frac{\langle \Phi(I^n)(\mathbf{p}), \Phi(\hat{I}^n)(\mathbf{p}) \rangle}{\|\Phi(I^n)(\mathbf{p})\|_2 \cdot \|\Phi(\hat{I}^n)(\mathbf{p})\|_2}. \tag{23}$$

Pixels with low similarity indicate semantic inconsistency, which strongly correlates with dynamics or misalignment. We therefore define a pseudo-label:

$$m^*(\mathbf{p}) = \mathbb{I}(s(\mathbf{p}) < \tau_s), \tag{24}$$

and train f_θ with a binary cross-entropy loss:

$$\mathcal{L}_{\text{mask}} = -\sum_{\mathbf{p}} \left[m^*(\mathbf{p}) \log m(\mathbf{p}) + (1 - m^*(\mathbf{p})) \log(1 - m(\mathbf{p})) \right]. \tag{25}$$

The learned mask $m(\mathbf{p})$ is image-specific, capturing transient or local motion, while the static probabilities γ_a from section 3.3 encode global 3D geometric stability. Instead of merging them, we use them complementarily: γ_a governs Gaussian initialization and pruning in 3D space, whereas $m(\mathbf{p})$ serves as a soft weighting factor during rendering. This separation ensures robustness to both long-term transient outliers and short-term view-dependent inconsistencies.

Algorithm 2 Transient-Aware Mask Prediction

Require: Images $\{I^n\}$, depths $\{D^n\}$, poses $\{P_n\}$, GMFlow, DINOv2 features $\Phi(\cdot)$
Ensure: Per-pixel transient mask $m(\mathbf{p})$

For each image pair (I^n, I^m)**:**
  Compute rigid projection:
    $\hat{\mathbf{p}}' = \pi(R_{n,m}\pi^{-1}(\mathbf{p}, D^n(\mathbf{p})) + t_{n,m})$
  Rigid flow: $\hat{\mathbf{f}}(\mathbf{p}) = \hat{\mathbf{p}}' - \mathbf{p}$
  Observed flow: $\mathbf{f}(\mathbf{p}) = \text{GMFlow}(I^n, I^m)$
  Residual: $r(\mathbf{p}) = \|\mathbf{f}(\mathbf{p}) - \hat{\mathbf{f}}(\mathbf{p})\|_2$
Motion–Depth Feature Fusion:
  Construct descriptor:
    $F_{\text{dyn}}(\mathbf{p}) = \big[\, r(\mathbf{p}), \|\mathbf{f}(\mathbf{p})\|, \angle\mathbf{f}(\mathbf{p}), \tilde{D}^n(\mathbf{p}), \nabla D^n(\mathbf{p}) \,\big]$
  Predict transient probability: $m(\mathbf{p}) = f_\theta(F_{\text{dyn}}(\mathbf{p}))$
Semantic Consistency Supervision:
  Extract DINOv2 features $\Phi(I^n)$ and $\Phi(\hat{I}^n)$
  Compute similarity:
    $$s(\mathbf{p}) = \frac{\langle \Phi(I^n)(\mathbf{p}), \Phi(\hat{I}^n)(\mathbf{p}) \rangle}{\|\Phi(I^n)(\mathbf{p})\|_2 \, \|\Phi(\hat{I}^n)(\mathbf{p})\|_2}$$
  Pseudo label: $m^*(\mathbf{p}) = \mathbb{I}(s(\mathbf{p}) < \tau_s)$
  Loss:
    $\mathcal{L}_{\text{mask}} = -\sum_{\mathbf{p}} \big[m^* \log m + (1-m^*) \log(1-m) \big]$
  {notation m abbreviates $m(\mathbf{p})$}
return Learned transient mask $m(\mathbf{p})$

3.5. Refined 3D Gaussian Splatting Reconstruction

In the process of 3D reconstruction, we integrate both geometry-derived static probabilities, γ_a, and learned per-pixel masks, $m(\mathbf{p})$, into the 3DGS framework. Standard 3DGS assumes a fully static environment, which makes it vulnerable to fitting transient objects as persistent structures. Our transient-aware variant modifies initialization, pruning(Sec.3.5.1), and loss formulation (Sec.3.5.2) to explicitly account for transients.

3.5.1 Initialization & Pruning

We initialize Gaussian primitives only from SfM points with high static confidence ($\gamma_i > \tau_{\text{init}}$). For each Gaussian i, we set its initial opacity parameter $\alpha_i = \gamma_i$, ensuring that highly static points start with stronger influence in the rendering process. This biases optimization toward stable regions and prevents overfitting to transients.

During optimization, Gaussian primitives undergo periodic pruning to maintain efficiency. We extend this by introducing a survival score:

$$s_i = \gamma_i \cdot (1 - \bar{m}_i), \tag{26}$$

where $\bar{m}_i$ is computed by projecting Gaussian i into all views where it is visible, bilinearly sampling the per-view transient mask $m_{i,v}$ at the projected location, and averaging:

$$\bar{m}_i = \frac{1}{|\mathcal{V}_i|} \sum_{v \in \mathcal{V}_i} m_{i,v}. \tag{27}$$

This yields a robust estimate of how likely Gaussian i belongs to a transient region. Gaussians with $s_i < \tau_{\text{survive}}$ are immediately pruned, ensuring that transient regions are aggressively eliminated from the representation.

3.5.2 Optimization

To prevent transient content from dominating optimization, we introduce mask-weighted objectives. The rendering loss is modified as

$$\mathcal{L}_{\text{render}} = \sum_{\mathbf{p}} (1 - m(\mathbf{p})) \cdot \|\hat{C}(\mathbf{p}) - C(\mathbf{p})\|_1, \tag{28}$$

where $C(\mathbf{p})$ is the ground-truth color and $\hat{C}(\mathbf{p})$ is the rendered color. Pixels with high transient probability are down-weighted, forcing the model to focus on static regions.

In addition, we incorporate a semantic consistency term using DINOv2 features:

$$\mathcal{L}_{\text{semantic}} = \sum_{\mathbf{p}} (1 - m(\mathbf{p})) \cdot \big(1 - s(\mathbf{p})\big), \tag{29}$$

which encourages rendered images to match input images at a feature level in static areas while ignoring transients.

Furthermore, we add a mask-weighted differentiable SSIM loss to encourage perceptual similarity:

$$\mathcal{L}_{\text{D-SSIM}} = \sum_{\mathbf{p}} (1 - m(\mathbf{p})) \cdot \big(1 - \text{SSIM}(\hat{C}(\mathbf{p}), C(\mathbf{p}))\big), \tag{30}$$

where $\text{SSIM}(\cdot)$ denotes the structural similarity index computed in a local window around pixel $\mathbf{p}$. This term complements the ℓ_1 loss by penalizing structural distortions while ignoring transient regions.

The overall objective becomes

$$\mathcal{L} = \mathcal{L}_{\text{render}} + \lambda_{\text{D-SSIM}} \mathcal{L}_{\text{D-SSIM}} + \lambda_{\text{mask}} \mathcal{L}_{\text{mask}} + \lambda_{\text{sem}} \mathcal{L}_{\text{semantic}}, \tag{31}$$

where $\mathcal{L}_{\text{mask}}$ supervises the mask predictor, and the other terms directly optimize Gaussian parameters. Training alternates between updating Gaussians and refining the mask predictor. At inference, the learned masks $m(\mathbf{p})$ act solely as soft weights for rendering and pruning, while γ_i remains fixed from Section 3.3.

4. Experiments

4.1. Datasets

We evaluate our method on two representative datasets: **NeRF On-the-go** [19] and **RobustNeRF** [21]. The NeRF

Figure 3: Qualitative evaluation on the NeRF On-the-go dataset

Table 1: Quantitative results on the NeRF On-the-go dataset. The best results are highlighted in **bold**, and the second-best in underline. Evaluation metrics include PSNR, SSIM, and LPIPS.

	Fountain			Corner			Patio			Train Station			Mean		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
3DGS [8]	20.262	0.632	0.224	21.839	0.812	0.199	18.151	0.725	0.239	20.144	0.763	0.235	20.099	0.733	0.224
WildGaussians [9]	19.279	0.594	0.338	23.559	0.843	0.176	21.343	0.786	0.182	21.518	0.741	0.221	21.425	0.741	0.229
RobustSplat [4]	20.549	0.685	0.239	25.162	0.872	0.151	21.451	0.795	0.167	22.475	0.813	0.147	22.409	0.791	0.176
SpotlessSplat [20]	20.325	0.615	0.234	25.249	0.879	0.136	21.332	0.783	0.152	22.132	0.792	0.155	22.260	0.767	0.169
Ours	**20.983**	**0.679**	**0.201**	**25.963**	**0.892**	**0.115**	**21.689**	**0.843**	**0.125**	**23.163**	**0.852**	**0.113**	**22.950**	**0.816**	**0.138**

On-the-go dataset consists of real-world scenes captured using handheld mobile devices under diverse outdoor and indoor conditions. It includes complex lighting variations, motion blur, and transient disturbances (such as moving objects), making it a suitable benchmark for testing the robustness and generalization ability of neural radiance field methods in unconstrained environments. For evaluation, we select four commonly used scenes: *Fountain*, *Corner*, *Patio*, and *Train Station*.

On the other hand, the RobustNeRF dataset primarily fo-

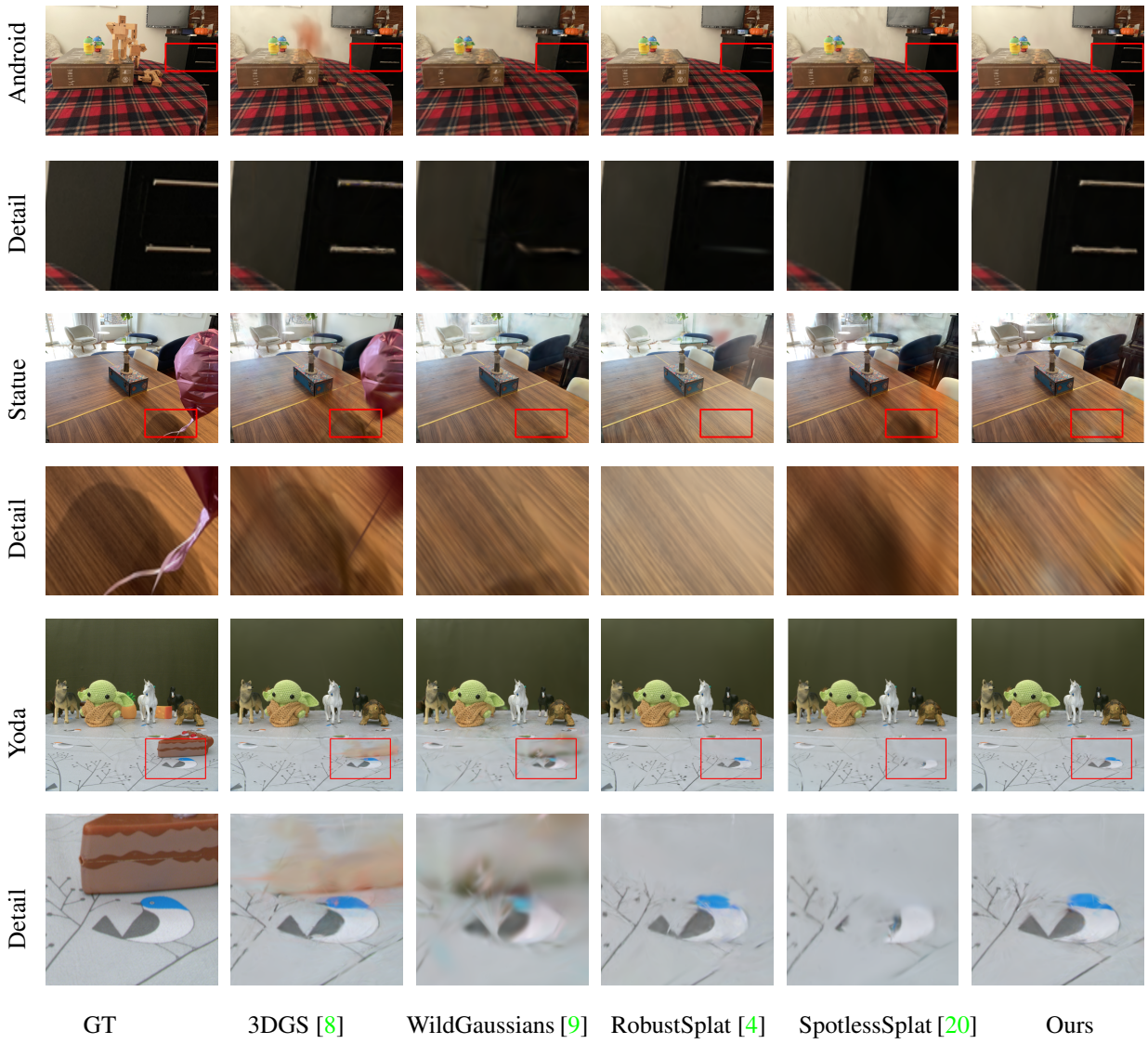

Figure 4: Qualitative evaluation on the RobustNeRF dataset

cuses on handling challenges such as occlusions, viewpoint inconsistencies, and degraded image quality, while also including transient disturbances, making it an ideal choice for evaluating reconstruction stability under imperfect input data. It provides a variety of synthetic and real scenes. For evaluation, we select four representative scenes: *Android*, *Statue*, *Crab2*, and *Yoda*. Together, these two datasets comprehensively assess the reconstruction fidelity and robustness of our proposed method in real-world scenarios containing transient disturbances.

4.2. Baselines

We conducted benchmark comparisons of our proposed method, specifically evaluating its performance against the vanilla 3D Gaussian Splatting [8], WildGaussians [9], RobustSplat [4], and SpotLessSplat [20]. Using both qualitative and quantitative metrics, we assessed each method's performance in terms of image reconstruction accuracy and detail preservation. The quantitative metrics include Peak Signal-to-Noise Ratio (PSNR, dB), Structural Similarity Index Measure (SSIM), and Learned Perceptual Image Patch Similarity (LPIPS), providing a comprehensive evaluation of the effectiveness of our method.

4.3. Implementation Details

Our implementation builds upon the official codebases of MASt3R [10] and 3D Gaussian Splatting [8] using PyTorch. All experiments were conducted on a single NVIDIA RTX

4090 GPU. An aggressive downsampling factor of 4 was applied to all images to maximize computational efficiency in per-pixel operations including motion flow estimation, feature extraction, and mask prediction, while maintaining sufficient detail for reliable reconstruction. We employed the pre-trained MASt3R model for initial pose and point cloud estimation. The EM algorithm ran for 3 iterations with static prior $\rho = 0.9$, providing stable motion segmentation across frames. For sparse scene graph construction, we used ASMK aggregation with a 64k codebook, which efficiently encodes local geometric features.

The transient-aware mask predictor f_θ was implemented as a lightweight 5-layer MLP with ReLU activations. It was trained using the Adam optimizer at a learning rate of $1e-3$ for 10k iterations with DINOv2-derived pseudo-labels and a confidence threshold of $\tau_s = 0.75$. For refined 3DGS reconstruction, Gaussians were initialized from points with $\gamma_i > 0.7$, with survival threshold $\tau_{\text{survive}} = 0.5$, ensuring that only reliable points contributed to the final representation. Training was performed with Adam for 30k iterations, using loss weights $\lambda_{\text{D-SSIM}} = 0.2$, $\lambda_{\text{mask}} = 0.1$, and $\lambda_{\text{sem}} = 0.05$ to balance structural, mask, and semantic consistency. Minor data augmentations, such as random rotations and color jittering, were applied to improve robustness without significantly increasing computational cost.

4.4. Comparison on the NeRF On-the-go Dataset

We comprehensively evaluate our proposed method on the NeRF On-the-go dataset [19] and perform comparative analysis with four state-of-the-art approaches. Figure 3 presents qualitative results from representative scenes containing transient disturbances such as pedestrians and vehicles, including flowing water in fountain scenes, moving pedestrians in street corner areas, and moving trains in railway station scenarios. While maintaining the integrity of the main scene structure, our method effectively suppresses interference from these transient elements and reconstructs clearer geometric details, particularly excelling in areas sensitive to transient noise such as edges and corners.

Quantitative comparisons are shown in Table 1.Our method achieves optimal metrics across all test scenes: highest PSNR and SSIM, and lowest LPIPS. In scenes with the most significant transient disturbances—street corners and railway stations—our method shows particularly notable advantages over the second-best approaches (SpotlessSplat and RobustSplat), with PSNR improvements exceeding 0.8 dB. This proves that our transient-aware mechanism effectively overcomes the impact of transient elements on reconstruction quality. Although comparative methods perform well in some static regions, they generally exhibit artifacts and detail blur when handling dynamic elements like moving pedestrians and vehicles. In contrast, through an explicit transient suppression strategy, our method effectively eliminates transient interference while preserving the integrity of the scene's static structure.

Experimental results demonstrate that TA-GS significantly outperforms existing methods in reconstruction accuracy and visual fidelity. Its core advantage lies in effectively distinguishing and handling static elements and transient disturbances within scenes, providing a more reliable solution for 3D reconstruction in dynamic environments.

4.5. Comparison on the RobustNeRF Dataset

We further evaluate TA-GS on the RobustNeRF dataset, which involves various transient interference scenarios such as moving pedestrians and vehicle traffic. Results are shown qualitatively in Figure 4 and quantitatively in Table 2.

As seen in Figure 4, TA-GS generates stable and artifact-free reconstructions across all scenes with significant transient interference. This is due to the combined effect of transient-aware mask learning, probabilistic static modeling, and mask-guided Gaussian splatting. The system effectively models transient elements, suppressing them while preserving the static scene structures. In contrast to baseline methods, which suffer from blurring and ghosting due to transient interference, TA-GS maintains geometric consistency and visual authenticity, even in regions affected by moving objects. It also accurately reconstructs fine textures of static structures despite the interference.

Quantitative results in Table 2 show that our method outperforms all other approaches with optimal scores: 28.73 PSNR, 0.898 SSIM, and 0.134 LPIPS. In scenes with the most severe transient interference, our method shows substantial improvement, proving the effectiveness of our transient processing framework. By jointly optimizing camera poses, 3D structure, static probabilities, and transient mask guidance, TA-GS efficiently suppresses transient elements while maintaining the static scene integrity.

Our method's design—combining transient suppression, static modeling, and mask-guided optimization—consistently delivers robust reconstructions in highly transient environments, outperforming existing methods in both visual quality and quantitative metrics.

4.6. Ablation Study

To evaluate the contribution of each key component in our framework, we conduct a comprehensive ablation study on the NeRF On-the-go dataset [19]. Table 3 summarizes the quantitative results, including PSNR, SSIM, and LPIPS, for each variant. Specifically, we examine the effects of (i) initialization and pruning of 3D Gaussians, (ii) mask supervision during transient-aware mask training, (iii) the use of learned masks in Gaussian splatting and rendering, and (iv) Gaussian pruning during optimization. By systematically disabling these modules, we assess their individual impact

Table 2: Quantitative results on the RobustNeRF dataset. The best results are highlighted in **bold**, and the second-best in underline. Evaluation metrics include PSNR, SSIM, and LPIPS.

	Android			Statue			Crab2			Yoda			Mean		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
3DGS [8]	23.539	0.802	0.167	21.134	0.837	0.151	30.570	0.914	0.179	28.143	0.843	0.226	25.847	0.849	0.181
WildGaussians [9]	24.491	0.835	0.145	22.679	0.861	0.126	30.816	0.922	0.182	30.321	0.904	0.189	27.078	0.881	0.161
RobustSplat [4]	24.632	0.841	0.133	22.236	0.843	0.143	33.328	0.935	0.167	33.554	0.932	0.158	28.434	0.888	0.150
SpotlessSplat [20]	24.387	0.828	0.155	22.316	0.849	0.134	33.211	0.927	0.152	33.439	0.929	0.164	28.338	0.883	0.151
Ours	**24.895**	**0.844**	**0.131**	**22.873**	**0.864**	**0.117**	**33.439**	**0.946**	**0.139**	**33.714**	**0.939**	**0.147**	**28.730**	**0.898**	**0.134**

Table 3: Ablation study results for each component of TA-GS.

Method / Ablation	**Init & Pruning**	**Mask Supervision**	**Mask Usage**	**Gaussian Pruning**	**PSNR**	**SSIM**	**LPIPS**
Full Method (Ours)	✓	✓	✓	✓	22.950	0.816	0.138
w/o Initialization & Pruning		✓	✓	✓	21.422	0.796	0.145
w/o Mask Supervision	✓		✓	✓	21.655	0.801	0.151
w/o Mask Usage (No Mask)	✓	✓		✓	20.493	0.743	0.172
w/o Gaussian Pruning in Optimization	✓	✓	✓		21.867	0.809	0.142

Table 4: Comparison of Training Time and Memory Usage with State-of-the-Art Methods

Method	**Training Time (min)**	**Memory Usage (GB)**
Full Method (Ours)	26	6.3
3D Gaussian Splatting [8]	41	5.2
WildGaussians [9]	89	7.4
RobustSplat [4]	67	6.9
SpotLessSplat [20]	72	8.5

on the removal of transient elements and overall reconstruction quality.

Additionally, we compare the training time and memory usage of our method with several state-of-the-art approaches, as shown in Table 4.

Efficiency of Initialization & Pruning: The initialization and pruning stage leverages static probabilities to initialize 3D Gaussian points and remove unlikely transient points early in the pipeline. Skipping this stage allows transient elements to persist, causing instability in fine structures. As shown in Table 3, PSNR drops by more than 1.5 points and SSIM decreases slightly, highlighting that early filtering is crucial for establishing a robust geometric foundation and reducing transient artifacts.

Efficiency of Mask Supervision: Mask supervision provides pixel-level guidance for training the transient-aware mask using motion–depth cues and semantic consistency. Without this supervision, transient regions are insufficiently suppressed, leading to ghosting and residual artifacts. As reported in Table 3, PSNR and SSIM decrease moderately while LPIPS increases, showing that mask supervision is essential for accurately identifying transient elements and improving reconstruction fidelity.

Efficiency of Mask Usage (No Mask): Applying the learned transient mask during Gaussian splatting and rendering downweights contributions from transient elements and guides pruning. When the mask is ignored, transient points influence the reconstruction unchecked, producing visible artifacts and reducing high-frequency detail. This demonstrates that mask usage plays a critical role in maintaining photorealistic reconstruction.

Efficiency of Gaussian Pruning in Optimization: Pruning during optimization removes points identified as transient based on static probabilities and mask guidance. If this step is disabled, residual transient points persist throughout training, leading to subtle blurring and remaining artifacts. Continuous pruning ensures a clean and stable reconstruction free from transient elements.

Visual Analysis of Component Synergy: Figure 5 presents a visual ablation study. As observed in column (b), omitting the Transient-Aware Mask leads to distinct 'ghosting' artifacts. In this case, although the geometry is partially cleaned, the optimization process forcibly fits transient textures onto the static background due to the lack of gradient shielding. Conversely, as shown in column (c), disabling the Probabilistic SfM results in geometric inconsistencies, such as floaters and incomplete background inpainting. This occurs because, without the probabilistic survival weights, transient points are erroneously densified early in training. Our full method effectively combines coarse geometric rejection with fine-grained pixel-level masking. This

synergy ensures that transient regions are handled consistently across both geometry initialization and appearance optimization, producing the cleanest reconstruction with sharp details."

(a) GT (b) w/o Mask (c) w/o SfM (d) Full

Figure 5: Visual ablation study demonstrating the synergy between components.

Training Efficiency and Memory Usage: Table 4 compares the training efficiency and memory requirements of our full method against several state-of-the-art Gaussian splatting approaches. The reported training time and memory usage are averaged over multiple scenes from the NeRF On-the-go dataset [19] to ensure a fair and consistent comparison. All experiments are conducted on 4× downsampled versions of the dataset, and the reported training time includes the structure-from-motion (SfM) preprocessing stage, providing an end-to-end measurement of the total training cost. Our method achieves the shortest training time of 26 minutes, which is approximately 37% faster than the 3D Gaussian Splatting baseline and more than 3× faster than WildGaussians [9]. In terms of memory consumption, our model requires 6.3 GB of GPU memory, remaining moderate compared with other approaches (e.g., WildGaussians 7.4 GB and SpotLessSplat 8.5 GB). This demonstrates that our method achieves a favorable trade-off between computational efficiency and memory usage, while maintaining competitive rendering quality.

Decoupling Motion and Semantics: To further investigate the complementary roles of visual cues, we evaluated variants of the mask predictor trained with decoupled features. As illustrated in Figure 6, visual inspection reveals that while this variant successfully captures actively moving objects, it fails to mask 'temporarily static' transient objects—such as vehicles waiting at a traffic light or pedestrians standing still—since their optical flow residuals are negligible. This leads to 'ghosting' artifacts where these objects are partially reconstructed into the scene. In contrast, the variant using only Semantic Consistency proves robust to motion states, successfully identifying static transient objects. However, due to the low spatial resolution of deep semantic features, this variant lacks pixel-level precision. It tends to generate coarse, dilated masks that oversegment the scene, erroneously removing static background details near the boundaries of transient objects. By fusing both modalities, TA-GS leverages motion cues for boundary precision and semantic cues for category-level robustness. This synergy achieves the highest fidelity, as evidenced by the optimal performance in all metrics.

(a) Motion (b) Semantic (c) Ours (d) GT

Figure 6: Visual comparison of predicted transient masks. (a) Motion cues alone fail to detect the stationary car. (b) Semantic cues alone detect the car but produce coarse, dilated boundaries. (c) Our fused approach achieves accurate and sharp masking, closely matching the Ground Truth (d)

.

Overall, the ablation study confirms that each module contributes in a complementary manner: initialization and pruning provide a stable geometric base, mask supervision improves transient detection, mask usage suppresses unwanted transient contributions, and Gaussian pruning refines the scene iteratively. Moreover, our method achieves competitive training time and moderate GPU memory usage, demonstrating that each module not only contributes to reconstruction quality but also maintains favorable computational efficiency.

5. Conclusion

We propose TA-GS, a framework for robust static scene reconstruction from image sequences with transient objects. Our approach introduces a geometry-first transient suppression paradigm, combining a probabilistic SfM formulation that jointly estimates camera poses, 3D structure, and point-level static probabilities via Expectation-Maximization, with a transient-aware mask learning module that leverages motion-depth cues and DINOv2 semantic consistency for accurate pixel-level transient region detection. These components are integrated into the 3D Gaussian Splatting framework, guiding Gaussian initialization, pruning, and rendering optimization. Extensive experiments show that TA-GS outperforms state-of-the-art methods in reconstruction quality and transient artifact suppression, producing cleaner geometry, sharper textures, and more faithful colors.

References

[1] S. Agarwal, K. Mierle, et al. Ceres solver: Tutorial & reference. *Google Inc*, 2(72):8, 2012. 6

[2] B. P. Duisterhof, L. Zust, P. Weinzaepfel, V. Leroy, Y. Cabon, and J. Revaud. Mast3r-sfm: a fully-integrated solution for unconstrained structure-from-motion. In *2025 International Conference on 3D Vision (3DV)*, pages 1–10. IEEE, 2025. 2, 4

[3] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 2

[4] C. Fu, Y. Zhang, K. Yao, G. Chen, Y. Xiong, C. Huang, S. Cui, and X. Cao. Robustsplat: Decoupling densification and dynamics for transient-free 3dgs. *arXiv preprint arXiv:2506.02751*, 2025. 2, 3, 9, 10, 12

[5] J. Guo, H. Xiao, and W. Kang. Ea-3dgs: Efficient and adaptive 3d gaussians with highly enhanced quality for outdoor scenes. *arXiv preprint arXiv:2505.10787*, 2025. 2

[6] P. Hedman, P. P. Srinivasan, B. Mildenhall, J. T. Barron, and P. Debevec. Baking neural radiance fields for real-time view synthesis. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5875–5884, 2021. 1

[7] Y. Jiang, C. Yu, T. Xie, X. Li, Y. Feng, H. Wang, M. Li, H. Lau, F. Gao, Y. Yang, et al. Vr-gs: A physical dynamics-aware interactive gaussian splatting system in virtual reality. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–1, 2024. 2

[8] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2, 3, 9, 10, 12

[9] J. Kulhanek, S. Peng, Z. Kukelova, M. Pollefeys, and T. Sattler. Wildgaussians: 3d gaussian splatting in the wild. *arXiv preprint arXiv:2407.08447*, 2024. 2, 3, 9, 10, 12, 13

[10] V. Leroy, Y. Cabon, and J. Revaud. Grounding image matching in 3d with mast3r. In *European Conference on Computer Vision*, pages 71–91. Springer, 2024. 2, 3, 4, 10

[11] C.-Y. Lin, Q. Fu, T. Merth, K. Yang, and A. Ranjan. Fastsr-nerf: Improving nerf efficiency on consumer devices with a simple super-resolution pipeline. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6036–6045, 2024. 1

[12] J. Lin, J. Gu, L. Fan, B. Wu, Y. Lou, R. Chen, L. Liu, and J. Ye. Hybridgs: Decoupling transients and statics with 2d and 3d gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 788–797, 2025. 3

[13] A. Markin, V. Pryadilshchikov, A. Komarichev, R. Rakhimov, P. Wonka, and E. Burnaev. T-3dgs: Removing transient objects for 3d scene reconstruction. *arXiv preprint arXiv:2412.00155*, 2024. 2, 3

[14] R. Martin-Brualla, N. Pandey, M. Khan, and J. Gall. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7210–7219, 2021. 3

[15] 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. *Communications of the ACM*, 65(1):99–106, 2021. 1

[16] T. Müller, A. Evans, C. Schied, and A. Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)*, 41(4):1–15, 2022. 1

[17] M. Niemeyer, J. T. Barron, B. Mildenhall, M. S. Sajjadi, A. Geiger, and N. Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5480–5490, 2022. 3

[18] W. Park, M. Nam, S. Kim, S. Jo, and S. Lee. Forestsplats: Deformable transient field for gaussian splatting in the wild. *arXiv preprint arXiv:2503.06179*, 2025. 3

[19] W. Ren, Z. Zhu, B. Sun, J. Chen, M. Pollefeys, and S. Peng. Nerf on-the-go: Exploiting uncertainty for distractor-free nerfs in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8931–8940, 2024. 3, 8, 11, 13

[20] S. Sabour, L. Goli, G. Kopanas, M. Matthews, D. Lagun, L. Guibas, A. Jacobson, D. Fleet, and A. Tagliasacchi. Spotlesssplats: Ignoring distractors in 3d gaussian splatting. *ACM Transactions on Graphics*, 44(2):1–11, 2025. 2, 3, 9, 10, 12

[21] S. Sabour, S. Vora, D. Duckworth, I. Krasin, D. J. Fleet, and A. Tagliasacchi. Robustnerf: Ignoring distractors with robust losses. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20626–20636, 2023. 3, 8

[22] J. L. Schonberger and J.-M. Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 2

[23] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006. 2

[24] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024. 2

[25] C. Wu et al. Visualsfm: A visual structure from motion system, 2011. 2

[26] H. Xu, J. Zhang, J. Cai, H. Rezatofighi, and D. Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8121–8130, 2022. 7

[27] V. Ye, R. Li, J. Kerr, M. Turkulainen, B. Yi, Z. Pan, O. Seiskari, J. Ye, J. Hu, M. Tancik, et al. gsplat: An open-source library for gaussian splatting. *Journal of Machine Learning Research*, 26(34):1–17, 2025. 2

[28] Y. Zhang, W. Jia, W. Niu, and M. Yin. Gaussianspa: An" optimizing-sparsifying" simplification framework for compact and high-quality 3d gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26673–26682, 2025. 2