Pattern Recognition 177 (2026) 113341

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

MFGS: Mask-free Gaussian separation for 3D object reconstruction

Jinguang Tong [a,b,*], Xuesong Li [a,b], Sundaram Muthu [b,c], Fahira Afzal Maken [b], Lars Petersson [b], Chuong Nguyen [a,b], Hongdong Li [a]

[a] Australian National University, The Australian National University, Canberra, 2601, ACT, Australia
[b] CSIRO, Black Mountain, Canberra, 2601, ACT, Australia
[c] Department of Data Science and Engineering, IISER, Bhopal Bypass Road, Bhopal, 462066, Madhya Pradesh, India

ARTICLE INFO

Keywords:
3D reconstruction
Gaussian splatting
Self-supervised learning

ABSTRACT

Accurate 3D reconstruction from multi-view images is a fundamental problem in computer vision. A common acquisition strategy involves placing an object on a rotating turntable while moving the camera to capture it from various viewpoints. In such scenarios, object moves relative to the background, many existing reconstruction methods rely on object masks to separate the foreground from the background. The quality of these masks significantly affects the final reconstruction, yet obtaining high-quality and consistent masks is a challenging and laborious process, especially when controlled environments like green screens are unavailable. To address this limitation, we introduce Mask-free Gaussian Separation (MFGS), a novel method that performs simultaneous object reconstruction and segmentation without requiring any input masks. Our approach builds on Gaussian Splatting and automatically disentangles the scene by extending each Gaussian primitive with a learnable parameter that represents its probability of belonging to the dynamic foreground object. This separation is optimized in a self-supervised manner, optimized by the object and camera transformation constraints. We evaluated MFGS on new synthetic and real-world datasets designed to reflect this challenging capture scenario. Experimental results demonstrate that our mask-free approach significantly outperforms existing methods. Notably, MFGS surpasses the performance of the state-of-the-art method(2DGS) that relies on high-quality segmentation masks, achieving a 27% improvement in novel view synthesis and a 7% improvement in geometry reconstruction.

1. Introduction

Accurate 3D modeling from images is a long-standing challenge in computer vision and graphics [1], supporting key applications in virtual reality, robotics, and digital manufacturing. Modern reconstruction techniques-most notably Neural Radiance Fields (NeRFs) [2–6] and Gaussian Splatting (3DGS) [7–11] have greatly advanced high-fidelity scene representation and synthesis.

For object-level reconstruction, dense multi-view capture remains essential, and turntable-based systems [12,13] provide a practical and widely used solution due to their accuracy, repeatability, and high throughput. In these systems, the object rotates around its vertical axis while the camera moves across several elevation angles, yielding a near-hemispherical image distribution. Although this setup is common in industrial inspection, cultural-heritage digitization, and commercial scanning platforms [12,14], it introduces a fundamental challenge for existing reconstruction algorithms: the object moves relative to the background. As a result, the foreground object and background cannot be treated as a whole, and standard multi-view assumptions are violated.

To cope with this, most methods rely on an external segmentation pipeline to mask out the background. However, obtaining accurate masks is often difficult in practice; cluttered environments, complex object shapes, thin structures, and partial glossy appearance frequently cause mask leakage or missing regions. This dependency significantly limits the applicability of existing NeRF- and GS-based approaches in real-world conditions, where masks are inaccurate, labor-intensive to annotate, or simply unavailable.

To address this limitation, we formulate a new and practical problem setting: object-centric 3D reconstruction from turntable scans in uncontrolled environments, without requiring segmentation masks. This setting reflects real digitization workflows where precise background control is rarely feasible.

Fig. 1 (left) illustrates the challenges posed by an unstructured capture environment. To overcome this bottleneck, we propose a mask-free reconstruction framework that jointly separates the object from the background while optimizing the Gaussian representation. Our key idea is to introduce a learnable probability for each Gaussian primitive, allowing the model to infer whether it belongs to the object or the

* Corresponding author.
E-mail address: hirotong0715@gmail.com (J. Tong).

https://doi.org/10.1016/j.patcog.2026.113341
Received 3 July 2025; Received in revised form 17 February 2026; Accepted 18 February 2026
Available online 20 February 2026

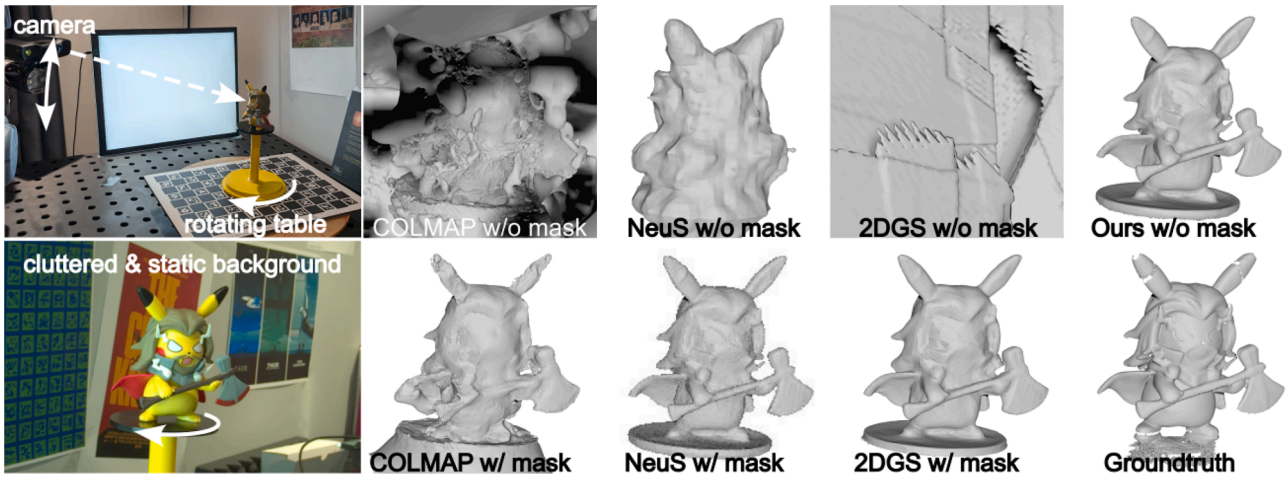


Fig. 1. Conventional rotating turntable setup under an unstructured capture environment. Existing methods fail to reconstruct the object without background removal. Even with object masks from SAM [16], performance remains inferior to our mask-free approach (MFGS), as the cluttered background and fine structural gaps pose significant challenges for accurate segmentation.

background. This enables reliable, self-supervised foreground-background separation directly from multi-view photometric cues, without requiring external masks or manual preprocessing. By removing this fragile dependency on segmentation, our approach significantly improves the robustness and practicality of 3D reconstruction in unconstrained turntable setups.

The contributions of this paper are as follows:

- We formally define the problem of 3D object reconstruction in turntable-based scanning setups, where both the object and background undergo relative motion. This setting reflects real-world industrial and cultural-heritage digitization scenarios and removes the impractical assumption of perfect segmentation masks.
- We propose MFGS, a novel self-supervised framework that jointly performs mask-free object-background separation and accurate 3D reconstruction. Our key idea is to equip each Gaussian primitive with a learnable foreground probability, enabling robust geometric recovery without external masks.
- We introduce new synthetic and real-world datasets tailored to this challenging scanning scenario, providing the first benchmark for evaluating reconstruction methods under turntable motion with complex, cluttered backgrounds.

This paper is organized as follows: Section 2 introduces the related work regarding 3D reconstruction using neural radiance field as well as Gaussian splat methods. Section 3 describes the 2D Gaussian Splatting method [15] upon which our method is based. Section 4 details our methodology. The empirical evaluation of our method and comparison with the state-of-the-art are provided in Section 5. Finally, Section 6 concludes the paper and discusses future work.

2. Related work

2.1. Neural 3D reconstruction

NeRF [2] can render high-fidelity novel-view images, but it performs poorly at extracting accurate 3D surfaces from the volume density field. To address the limitation, neural implicit functions such as signed distance function (SDF) [17] and occupancy grids [18,19] are usually preferred to define 3D surfaces due to their ability to represent smooth and precise geometry. Wang et al. [4] adopt an SDF representation and convert the signed distance value into density for rendering novel-view images via volume rendering, which enables surface reconstruction during view rendering [20]. To further enhance 3D reconstruction via novel-view synthesis, Fu et al. [3] employ sparse 3D points, generated by Structure from Motion, to explicitly supervise the SDF network in learning 3D surfaces. Additionally, the method incorporates multi-view photometric consistency constraints to further refine and enhance the accuracy of 3D reconstruction. MonoSDF [21] leverages monocular geometric cues, such as monocular depth and normals, to refine the geometric reconstruction beyond typical rendering loss. To accelerate neural surface reconstruction, NeuS2 [5] parameterizes SDF using multi-resolution hash tables of learnable feature vectors, handling high spatial resolution with reduced computational cost. Neuralangelo [22] leverages the power of multi-resolution 3D hash grids [23] and neural surface rendering to address challenges in reconstructing detailed structures from real-world scenes. This approach offers a scalable solution for high-fidelity surface reconstruction from RGB images without auxiliary data like segmentation or depth maps. By employing a progressive optimization strategy using coarse-to-fine hash grids, Neuralangelo captures intricate details in the reconstructed geometry. Additionally, it introduces a curvature loss to promote smooth surfaces, further enhancing the overall quality of the reconstructed geometry. Recent works have also explored under-constrained 3D surface reconstruction pipelines [24,25], as well as generative priors for sparse-view geometry estimation [26]. Despite advancements in neural 3D surface reconstruction techniques [4–6,22,23,27–29], challenges persist in their efficient training and rendering. These neural implicit surfaces are primarily designed for static scenes and face difficulties in adapting to dynamic scenes.

2.2. Gaussian splatting for 3D reconstruction

Recently, 3DGS [7] has shown impressive capabilities in novel-view synthesis with real-time rendering, and has been extended to reconstruct 3D surface from multi-view images [8,10,30]. By enforcing regularization terms to promote flat and well-distributed Gaussians with limited overlap, SuGaR [8] has developed an approach that efficiently extracts accurate and editable meshes while enhancing rendering quality. 3DGSR [31] integrates SDF with 3D Gaussians with a differentiable SDF-to-opacity transformation function, which transforms SDF values into Gaussian opacities, linking SDFs and Gaussians to enforce surface constraints and enable unified optimization. The Gaussian surfels [9] method flattens 3D Gaussian points into 2D ellipses to resolve normal ambiguity and improve surface alignment. Additionally, the method incorporates a self-supervised normal-depth consistency loss to ensure that the local z-axis aligns with the surface normal derived from rendered

depth maps. 2D Gaussian Splatting (2DGS) [15] uses 2D Gaussian primitives to model and reconstruct geometrically accurate radiance fields, in which a perspective-accurate 2D splatting process utilizes ray-splat intersection and rasterization. This approach helps in accurately recovering thin surfaces and achieving stable optimization. In this work, we aim to extend the capabilities of 2DGS for 3D surface reconstruction in mask-free scenario.

2.3. 4D dynamic Gaussians

In the proposed problem, where both the object and background are moving, forming a dynamic scene reconstruction problem. The representation of 3D Gaussians has also been adapted to reconstruct the geometrical structure and appearance of a dynamic scene [27,32–35]. Dynamic 3DGS [32] is a pioneering approach that extends 3DGS to a dynamic environment. It performs frame-by-frame optimization iteratively, effectively handling multi-view dynamic scenes with significant motion. Deformable 3DGS [33] proposes to learn a spatial-temporal deformation model (i.e. MLP) to map time-varying 3D Gaussians into a canonical space, along with a set of Gaussian in the canonical space, both of which are jointly optimized during volume rendering. Differently, 4DGS [36] modelled the deformation in a dynamic scene with multi-resolution voxel planes and a lightweight multi-head deformation decoder to further enhance the efficiency. Real-time dynamic reconstruction by [37] introduces a spatial-temporal volume representation using 4D Gaussian primitives that encode both geometry and appearance. These primitives utilize parameterizations based on anisotropic ellipses and 4D spherical harmonics. To improve the modeling of dynamic scene geometry, motion-flow-based 3DGS [34,35] introduces the optical flow in the flow loss function to constraint the movement of 3G Gaussinas in 3D space. This approach allows the motion offsets of 3D Gaussians to be splatted and rendered into optical flow images. However, these dynamic 3DGS approaches focus more on modelling time-varying appearance and geometry and fail to reconstruct accurate 3D structures.

2.4. 3D Gaussian-splatting segmentation

3DGS-CD [38] is presented as the first 3DGS-based method to detect object rearrangement. It fuses multiview 2D masks generated by EfficientSAM [39] and relies on 2D change detection and object association to obtain pose change and 3D segmentation. In contrast, we aim to segment foreground objects in 3D without 2D masks or 2D change detection, therefore eliminating human inputs required by models like SAM [39,40]. Feature-based approaches augment each Gaussian with learnable attributes. Gaussian Grouping [41] learns identity encodings, optimized with 2D SAM mask supervision and 3D consistency. Similarly, SAGA (Segment Any 3D Gaussians) [42] uses a scale-gated affinity feature per Gaussian, distilling 2D mask capabilities and handling multi-granularity segmentation. Existing methods often necessitate supplementary image encoders or segmentation masks. In contrast, our MFGS is entirely self-supervised, eliminating such additional dependencies.

3. Preliminary: 2D Gaussian splatting

Our method leverages the advanced geometry performance and efficiency of surfel-based 2D Gaussian Splatting (2DGS) [15] to achieve high-quality reconstructions. 2DGS collapses a 3D volume into 2D-oriented planar Gaussian disks and introduces a perspective-accurate 2D splatting process. Similar to 3DGS, each 2D splat is defined by its central point p_k, two principal tangential vectors t_u and t_v, and a corresponding scaling vector $S = (s_u, s_v)$ that controls the 2D Gaussian distribution's variances. Compared to 3DGS, 2DGS better represents scene geometry because its oriented planar Gaussians can align perfectly with surfaces, providing a well-defined normal direction.

Specifically, a 2D Gaussian disk is parameterized in a local tangent space within world coordinates as:

$$P(u, v) = \mathbf{x} + s_u \mathbf{r}_u u + s_v \mathbf{r}_v v \tag{1}$$

For a point $\mathbf{u}(u, v)$ in the uv space, its 2D Gaussian value is calculated by the standard Gaussian:

$$\mathcal{G}(\mathbf{u}) = \exp\left(-\frac{u^2 + v^2}{2}\right) \tag{2}$$

The center $\mathbf{x}$, scaling (s_u, s_v), and the rotation $(\mathbf{r}_u, \mathbf{r}_v)$ are all learnable parameters. Following 3DGS [7], each 2D Gaussian primitive also possesses an opacity α and view-dependent appearance c, parameterized using spherical harmonics Fig. 2. Instead of projecting the 2D Gaussian primitives directly onto the image plane for rendering, 2DGS determines the intersection point of a ray and a splat within the local tangent space by plane intersection, which alleviates the problem of splat degeneration, especially at grazing angles. The rasterization process is similar to 3DGS, in that 2D Gaussians are sorted based on the depth of their center and volumetric alpha blending is used to integrate alpha-weighted appearance from front to back:

$$\mathbf{c}(\mathbf{x}) = \sum_{i=1} \mathbf{c}_i \alpha_i \hat{G}_i(\mathbf{u}(\mathbf{x})) \prod_{j=1}^{i-1} (1 - \alpha_j \hat{G}_j(\mathbf{u}(\mathbf{x}))) \tag{3}$$

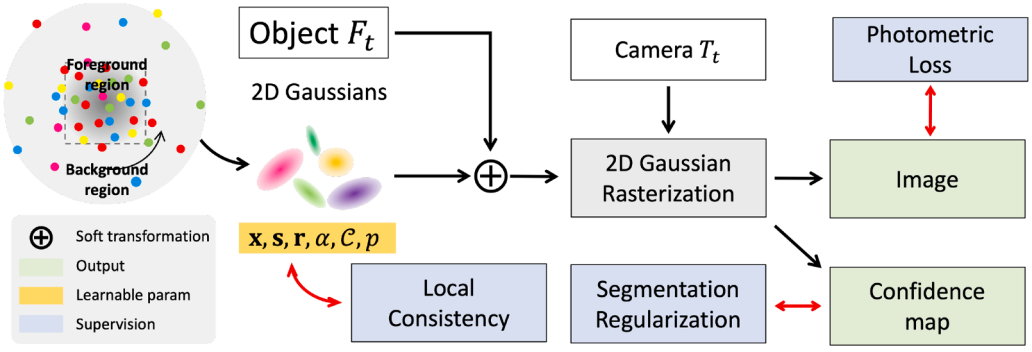


Fig. 2. Overview of our proposed MFGS pipeline. A canonical set of 2D Gaussians is initialized in the foreground and background regions. The 2D Gaussians are parameterized by the 3D location, 2D scale, 3D rotation, transparency, appearance and foreground probability $\{\mathbf{x}, \mathbf{s}, \mathbf{r}, \alpha, \mathbf{C}, p\}$. For each time step t, Gaussians are transformed into the canonical frame using a differentiable soft transformation based on the object motion F_t and the learned foreground probability p. Given camera pose T_t, the transformed Gaussians are rendered by 2D Gaussian rasterizer to produce an image and a confidence map. Local consistency regularization is applied in the canonical Gaussian space, while segmentation regularization supervises the confidence map and a photometric loss supervises the rendered image. Yellow elements denote learnable parameters and blue boxes indicate supervision signals. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4. Method

In this section, we introduce MFGS, a mask-free framework for reconstructing 3D objects in turntable-based settings with complex and cluttered backgrounds. Unlike conventional approaches that depend on accurate foreground masks to isolate the object from the background, our method removes this requirement entirely. At the core of MFGS is a fully self-supervised Gaussian separation mechanism that enables the model to automatically distinguish object and background components during optimization, allowing robust reconstruction.

4.1. Problem formulation

High-precision 3D reconstruction is essential in industrial inspection, cultural-heritage digitization, and commercial scanning systems. In turntable-based scanning setups, the object moves relative to the background, causing the foreground and background to follow different movements in the camera coordinate system. As a result, current methods [4,15,19] struggle because they implicitly assume a static background and therefore require accurate segmentation masks to isolate the object before reconstruction. In real production environments, obtaining such masks is highly nontrivial. Turntable systems are typically surrounded by clutter-robotic arms, fixtures, cables, and support structures-that produce complex and visually similar backgrounds. Even with attempts to clean the background, shadows, lighting variations, and environmental contamination often degrade segmentation quality. While green screens can help, they frequently obstruct camera motion or mechanical actuation [13,14], limiting their practicality in many real-world workflows.

Therefore, there is a need for a more robust method that can effectively handle background-foreground separation without relying on input masks. These challenges highlight the need for a reconstruction method that can reliably reconstruct the object without relying on externally supplied masks.

As illustrated in Fig. 1, we focus on reconstructing objects from multi-view images captured as they rotate on a turntable under varying camera elevations. In such setups, common in industrial and cultural-heritage digitization, the background cannot be easily controlled or isolated, and segmentation errors propagate directly into geometric artifacts. Thus, robust 3D reconstruction in this scenario requires explicitly addressing the difficulty of foreground-background separation under relative motion, without assuming access to accurate masks.

To address these challenges, we explicitly decompose the scene into two regions: a dynamic foreground, containing the rigid object rotating with the turntable, and a static background, containing all stationary elements in the environment. This separation reflects the physical structure of turntable-based scanning and provides a foundation for disentangling the two motion patterns.

To establish a consistent reference frame for reconstruction, we first define a canonical coordinate system, which corresponds to the background and object configuration at the initial time step. Given a set of N multi-view images $\mathcal{I} = \{I_t \mid t = 1, 2, \dots, N\}$, captured at discrete time steps, we describe the motion of both the camera and the rotating object with respect to this canonical system.

The camera poses are represented by $\mathcal{T} = \{T_t \in \mathbb{R}^{4\times4} \mid t = 1, 2, \dots, N\}$, where each T_t maps coordinates from the camera frame to the canonical system:

$$X_t = T_t \cdot x^c \tag{4}$$

where x^c denotes homogeneous coordinates in the camera's coordinate system.

The moving foreground object is described by a sequence of SE(3) [43] transformations $\mathcal{F} = \{F_t \in \mathbb{R}^{4\times4} \mid t = 1, 2, \dots, N\}$, each mapping points from the object's coordinate frame to the canonical coordinate system:

$$X_t = F_t \cdot x^f. \tag{5}$$

where x^f represents homogeneous coordinates in the foreground object's coordinate system. Because the object is assumed rigid, x^f is invariant across time.

These explicit definitions highlight the key distinction between our setting and standard multi-view 3D reconstruction: in typical scenarios, the object is static relative to the background, whereas in turntable-based scanning, the object and background undergo different motions. This mismatch breaks the assumptions of most existing pipelines, making reconstruction impossible unless the foreground object is cleanly separated from the moving background. Camera poses T_t and object motions F_t can be obtained either through structure-from-motion (SfM) [44] on the background and object separately or via system calibration in structured scanning setups.

A naïve solution is to segment each image using segmentation method. Although SAM [40] and its variants [16,45] can produce high-quality masks, they still require user prompts to operate reliably in real-world environments. This manual intervention prevents full automation and limits scalability, and even with careful prompting, obtaining consistent pixel-level accurate masks across all viewpoints remains a significant practical challenge.

4.2. Extended 2D Gaussian

As analyzed in Section 4.1, a key challenge of this setting is to reliably distinguish between the static background and the dynamically rotating object so that each can be processed using the correct transformation. This section introduces our fully self-supervised approach for achieving this separation as shown in Fig. 2.

This explicit, point-based representation is particularly well suited to our problem: unlike neural implicit fields such as NeRFs [2,46] or neural SDFs [4,5,19], 2D Gaussians provide fine-grained control over individual primitives. This allows us to assign and optimize custom attributes for each Gaussian individually, an ability that is crucial for modeling separate foreground and background behavior.

Following 2DGS [15], the scene is represented by a set of 2D Gaussian primitives, each parameterized by

$$\{\mathbf{x}_i, \mathbf{s}_i, \mathbf{r}_i, \alpha_i, C_i\}$$

where $\mathbf{x}_i \in \mathbb{R}^3$ is the Gaussian center, $\mathbf{s}_i \in \mathbb{R}^2$ its scale, $\mathbf{r}_i \in \mathbb{R}^4$ its rotation (quaternion), $\alpha_i \in \mathbb{R}$ its opacity, and $C_i \in \mathbb{R}^k$ the spherical harmonic coefficients encoding color.

To distinguish static from dynamic components of the scene, we augment each Gaussian with an additional learnable indicator representing its type (foreground or background). A binary variable would be ideal conceptually but is non-differentiable and unsuitable for gradient-based optimization. Instead, we introduce a continuous probability $p \in [0, 1]$, where $p = 1$ indicates that the Gaussian belongs to the dynamic foreground, and $p = 0$ indicates the static background. This leads to the final parameterization of each Gaussian primitive as

$$\{\mathbf{x}_i, \mathbf{s}_i, \mathbf{r}_i, \alpha_i, C_i, p_i\}.$$

The value of p_i determines which transformation, foreground or static, is applied during rendering and allows the model to learn the separation of the scene regions in a fully self-supervised manner.

4.3. Self-supervised Gaussian separation

To achieve fully self-supervised Gaussian separation, we integrate the learnable foreground probability p_i directly into the rendering pipeline, enabling differentiable separation of static and dynamic regions.

As described in Section 4.1, the scene consists of a static background (canonical coordinate system) and a dynamic foreground object rotating

with the turntable. Each has its own local coordinate frame, and during rendering, Gaussian primitives must be placed in the appropriate world-space location according to whether they belong to the foreground or background.

Unlike standard Gaussian Splatting methods [7,15], which initialize all Gaussians directly in world coordinates. Foreground Gaussians are defined in the local object coordinate system, their positions are transformed into the canonical coordinate system through the corresponding transformation F_t.

For end-to-end optimization, the mapping from local to canonical coordinates must be differentiable. Therefore, instead of using a hard assignment (foreground or background), we use a soft transformation based on the foreground probability $p_i \in [0, 1]$, representing the likelihood that Gaussian i belongs to the rotating foreground.

For a Gaussian center $\mathbf{x}_i$, its canonical-space position at time t is:

$$X_i^t = p_i F_t \cdot \mathbf{x}_i + (1 - p_i) \cdot \mathbf{x}_i \tag{6}$$

This equation can be viewed as the expected transformed position under the probability of belonging to the foreground. Crucially, it is fully differentiable, allowing both the Gaussian parameters and the probabilities p_i to be optimized jointly via photometric loss, eliminating the need for object masks.

During training, the foreground probability p_i naturally converges toward binary values (0 or 1), leading to a clean separation of static and dynamic Gaussians. This forces the probabilistic transformation in Eq. (6) to degenerate into a deterministic assignment to either the foreground or background frame. This mechanism serves as an automatic method for separating the scene's Gaussians.

The soft expectation formulation above is suitable for Gaussian centers but inappropriate for rotation-dependent parameters such as the quaternion $\mathbf{r_i}$ and the spherical harmonic coefficients C_i. Linear interpolation within the SO(3) group is notoriously complex and can hinder optimization [47]. To avoid this, we apply a deterministic "hard" transformation for all rotation-related parameters, conditioned on the final converged value of the foreground probability p_i.

This "hard" transformation relies on a definitive classification of each Gaussian. We assign a Gaussian to the foreground if its foreground probability $p_i \geq \tau$, and to the background otherwise, using an empirically determined threshold of $\tau = 0.8$. The rotation-related parameters are then transformed using the corresponding transformation (F_t)

For a Gaussian assigned to the foreground, the rotation is updated using the rotational component $F_t^{3\times3}$

$$\mathbf{r}_i^t = \mathcal{R}^{-1}(F_t^{3\times3} \cdot \mathcal{R}(\mathbf{r}_i)) \tag{7}$$

where $\mathcal{R}$ denotes the conversion from a quaternion to a rotation matrix, and $F_t^{3\times3}$ is the 3×3 rotation matrix extracted from the transformation F_t. For background Gaussians, the rotation remains unchanged since the canonical background frame is static.

The spherical harmonic coefficients C_i, which encode view-dependent appearance, must likewise be expressed in the correct world-frame orientation. We apply the corresponding Wigner D-matrix [48] to rotate the coefficients according to the assigned transformation.

Given a rotation R, its Euler-angle parameterization yields a Wigner D-matrix that rotates the spherical harmonic basis. Algorithm 1 summarizes this process.

Algorithm 1: Rotate spherical harmonics with wigner D-matrix.

Input: Spherical harmonics Y_{lm}, Rotation matrix R and Euler angles (α, β, γ)
Output: Rotated spherical harmonics $\tilde{Y}_{lm}$
begin
  $(\alpha, \beta, \gamma) \leftarrow$ `RotationMatrix2EulerAngles(`R`)`
  foreach l *in degrees of spherical harmonics* **do**
    $D^l \leftarrow$ `ComputeWignerDMatrix(`l, α, β, γ`)`
    for $m = -l$ **to** l **do**
      $\tilde{Y}_{lm} \leftarrow \sum_{m'=-l}^{l} D^l_{mm'} Y_{lm'}$
    end
  end
  return $\tilde{Y}_{lm}$
end

4.4. Optimization

To train our model effectively in the absence of ground-truth masks, we propose a self-supervised optimization framework that combines photometric loss with three regularization terms. These regularizers are designed to guide the separation of static and dynamic regions by enforcing spatial and probabilistic consistency.

Local Consistency Loss. We assume that Gaussians belonging to the same region (foreground or background) are spatially coherent and exhibit similar foreground probabilities. To encourage this behavior, we introduce a local consistency loss inspired by [41], which promotes smoothness in the probability distribution across neighboring Gaussians. Specifically, for each Gaussian, we compute the Kullback-Leibler (KL) divergence between its foreground probability p_i and those of its k-nearest neighbors:

$$\mathcal{L}_{lc} = \frac{1}{k} \sum_{j=1}^{k} D_{KL}(p_i || p_j) \tag{8}$$

3D Separation Regularization. As defined in Section 4.2, the foreground probability p_i represents the likelihood that a Gaussian belongs to the dynamic object. During training, these probabilities should converge toward binary values (0 or 1). We enforce this through an entropy-based regularizer that penalizes uncertainty in the prediction:

$$\mathcal{L}_{3ds} = \frac{1}{k} \sum_{i=1}^{k} -\big(p_i \log(p_i) + (1 - p_i) \log(1 - p_i)\big) \tag{9}$$

2D Separation Regularization. In addition to the 3D domain, we also impose regularization in the 2D image space. A confidence map C is rendered by replacing the color $\mathbf{c}_i$ with the corresponding probability p_i in the rendering equation (see Eq. (3)). We then apply a similar entropy-based loss over the rendered confidence map:

$$\mathcal{L}_{2ds} = \mathbb{E}\big[-(C \log(C) + (1 - C) \log(1 - C))\big] \tag{10}$$

Overall Objective. Our final loss combines the above regularization terms with the original 2DGS losses $\mathcal{L}_{2dgs}$, which include photometric and normal consistency terms. The complete training objective is:

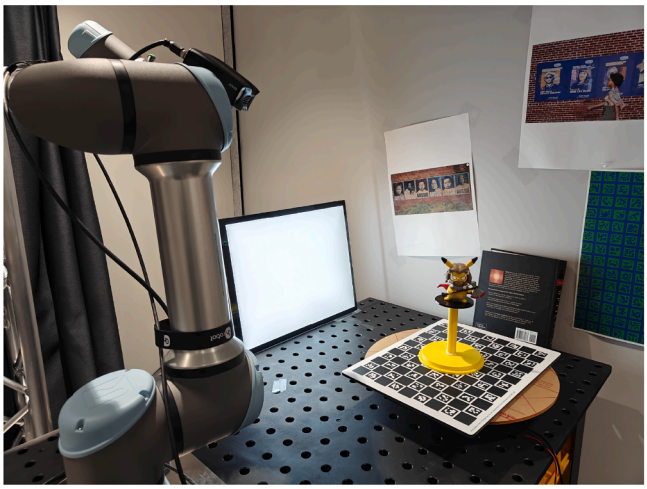
$$\mathcal{L} = \mathcal{L}_{2dgs} + \lambda_{lc}\mathcal{L}_{lc} + \lambda_{3ds}\mathcal{L}_{3ds} + \lambda_{2ds}\mathcal{L}_{2ds} \tag{11}$$

This composite loss enables robust and mask-free foreground-background separation while maintaining reconstruction fidelity.

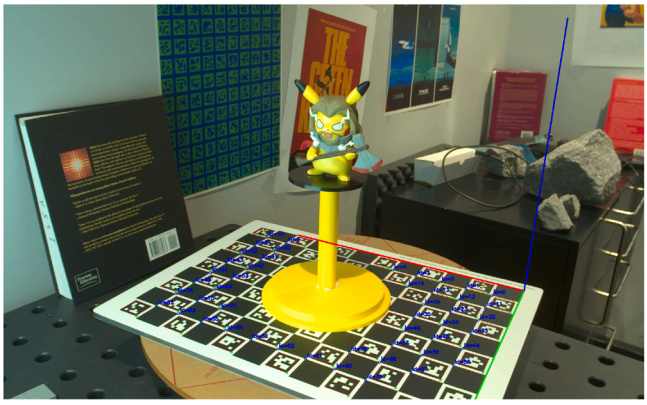
4.5. Implementation details

Initialization. To reduce manual effort, we avoid using Structure-from-Motion (SfM) points for initializing the Gaussian point cloud. Instead, we adopt a fully random initialization strategy. As previously described, foreground Gaussians are defined in their local coordinate systems. For the foreground region, we initialize Gaussians randomly within a cube centered at the origin. For the background region, Gaussians are randomly distributed on a surrounding sphere whose radius is four times the cube's side length. Foreground probabilities are initialized to $p_i = 0.9$ for Gaussians in the foreground and $p_i = 0.1$ for those in the background. All other parameters are initialized following the procedure described in the original 2DGS implementation [15].

Optimization. Our method is implemented using PyTorch and CUDA, and optimized using the Adam optimizer. All experiments are conducted



(a) Dataset capture setup.



(b) Estimated camera poses.

Fig. 3. Turntable-robot rig and pose estimation for the real dataset.

on a single NVIDIA RTX 4090 GPU. Following the training schedule of 2DGS, we train the model for a total of 30,000 iterations. For the real dataset, we set the loss weights as $\lambda_{lc} = 50$, $\lambda_{3ds} = 0.1$, and $\lambda_{2ds} = 0.1$; for the synthetic dataset, we use $\lambda_{lc} = 50$, $\lambda_{3ds} = 0.01$, and $\lambda_{2ds} = 0.01$. The regularization losses $\mathcal{L}_{lc}$, $\mathcal{L}_{3ds}$, and $\mathcal{L}_{2ds}$ are activated after the first 5000 iterations to allow stable initial convergence.

5. Experiments

5.1. Dataset details

Real dataset. We capture the real dataset in a laboratory environment using a carefully designed tabletop, object-centric rig (see Fig. 3a). A Zivid 2[1] RGB-D camera is mounted on the end-effector of a UR5e[2] arm, which allows us to place the camera at multiple, precisely controlled viewpoints around the object.

The object is fixed on a small stand above a motorized turntable, on which a ChArUco calibration board is mounted to provide accurate camera-world and object-world calibration. Printed posters and books are arranged in the background to create realistic clutter and texture, yielding challenging yet controlled scenes.

We carefully select 9 objects spanning a variety of shapes and materials and place in realistic tabletop scenes, introducing challenges such as background clutter and ambiguous object boundaries for mask-based methods.

During acquisition, each object is observed from five or seven elevation angles. For each fixed camera viewpoint, we rotate the object on the turntable and capture RGB and depth images every 10° at a resolution of 1944×1200 pixels. To match the pinhole camera model assumed by 2DGS, we crop the images using the calibrated intrinsics and downsample twice to 944×560.

Camera and object poses are estimated using the ChArUco board. We first calibrate the camera pose T_t with respect to the board at the canonical origin position. Then, we estimate the object transformation F_t relative to the same origin, yielding consistent camera and object poses across all views. In practical object-scanning systems, the object transformation and the camera poses can be calibrated once and reused for subsequent captures.

Ground-truth meshes are reconstructed via TSDF fusion [49] using Open3D [50] with a voxel size of $0.3mm$ and a truncation distance of $1.5mm$.

Synthetic dataset. Using Blender Cycles [51], we rendered nine objects across three carefully designed scenarios containing textured walls, furniture, and cluttered backgrounds. The objects cover a range of shapes and materials (e.g., matte surfaces, thin structures), which produce challenging cases for mask-based methods.

To render the dataset, cameras are placed at five elevation angles uniformly distributed in $(-30°, 30°)$, while the object rotates by 10° per capture, yielding 180 images per object at 800×800 resolution.

From the renderer, we also obtain ground-truth geometry and pixel-perfect object masks. These masks are used only for the 2DGS (GT) baseline and for evaluation, but are never used by our method.

For both real and synthetic dataset, we use 60% of the images for training and 20% each for validation and testing. Examples of the captured real dataset and the synthetic dataset are shown in Fig. 4, illustrating the variety of viewpoints, lighting, and background clutter in this dataset. We also include example images and a video of the full dataset in the supplementary material.

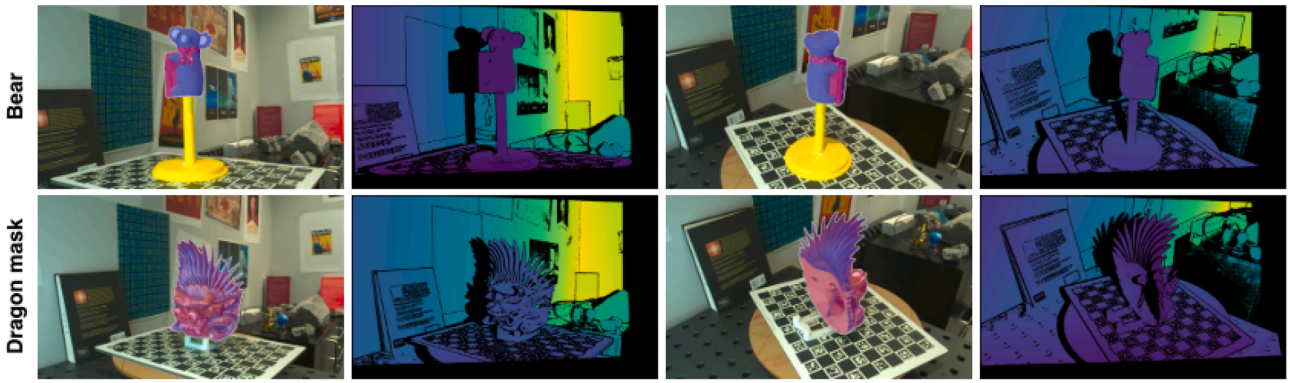
Cultural heritage dataset. Furthermore, to demonstrate the practical applicability of our method in real-world scenarios-such as cultural heritage preservation [52,53], we additionally captured two high-fidelity wooden bust sculptures, *Young Man* and *The Janger Dancer* as shown in Fig. 11. These objects were scanned using the same robotic arm and rotating turntable setup described earlier. Importantly, this dataset presents even greater reconstruction challenges: the wooden surfaces exhibit subtle glossiness, contain large low-texture regions and fine structures, both of which are known to degrade the performance of reconstruction methods due to view-dependent reflections and insufficient photometric cues. This additional dataset is used to validate the robustness of our approach in real-world digitization scenarios.

5.2. Experimental settings

Baselines. To evaluate the effectiveness of our method, we compare against four baseline approaches representative of different paradigms in 3D reconstruction.

COLMAP [44] is a traditional structure-from-motion pipeline that reconstructs 3D geometry from 2D images via feature matching and bundle adjustment. **NeuS** [4] integrates Signed Distance Functions (SDFs) within the NeRF framework to produce accurate surface reconstructions and photorealistic novel-view synthesis. **2DGS** [15] employs 2D Gaussian primitives to represent radiance fields, enabling improved surface alignment and real-time rendering with high geometric fidelity. **Deformable 3DGS (D-3DGS)** [33] extends 3D Gaussian Splatting to dynamic scenes by learning temporal deformations of both geometry and appearance, making it suitable for motion-rich environments. These

[1] https://www.zivid.com/zivid-2
[2] https://www.universal-robots.com/products/ur5-robot/



(a) Real dataset.

(b) Synthetic dataset.

Fig. 4. Examples of (a) the real dataset and (b) the synthetic dataset. The real dataset features cluttered tabletop backgrounds, varying materials, and fine object structures. The synthetic dataset provides controlled yet diverse object shapes and motions, enabling systematic evaluation of our mask-free reconstruction framework.

Table 1
Chamfer-$\mathcal{L}_1 \downarrow$ (scaled by 100×) on the real dataset. Cells are shaded best, second, third. "–" indicates the method failed to produce a mesh.

	Method	Bear	Captain	Controller	Dmask	Dog	Dragon	Pikachu	Plant	Rooster	Avg.
w/ mask	COLMAP	0.0663	0.2442	0.2246	0.2089	0.1656	4.2967	0.4280	16.7652	0.9057	2.5895
	NeuS	0.0695	0.1082	0.1484	0.1804	0.0964	0.1247	0.2990	0.1707	0.1839	0.1535
	2DGS	0.0690	0.1153	0.1815	0.1366	0.1063	0.1054	0.2514	0.1458	0.1170	0.1365
w/o mask	COLMAP	34.6392	45.4321	44.8083	35.4063	35.6062	32.4344	35.1575	41.1090	33.0436	37.5152
	NeuS	1.5605	–	1.4726	2.4949	–	2.4661	2.5834	2.2161	–	2.1323
	2DGS	26.4271	12.4420	23.9106	18.7352	–	20.5928	26.6814	24.6871	26.1505	22.4533
	D-3DGS	2.1397	1.6192	1.9330	1.5135	0.5591	6.3222	0.7764	6.1672	1.8238	2.5393
	MFGS	0.0769	0.1210	0.1166	0.1075	0.0860	0.1044	0.2966	0.1249	0.1111	0.1272

baselines offer a comprehensive benchmark across classical, implicit, and point-based reconstruction methods.

For the first three baselines (COLMAP, NeuS, and 2DGS), we conduct two variants of experiments: one using object masks and the other without. Object masks are generated using recent segmentation and tracking models, including Segment Anything and its variants [16,40,54]. Examples of these masks are illustrated in Fig. 7.

Same pose inputs. To ensure a fair comparison, we provide *the same calibrated pose inputs* as our method to all baselines whenever camera extrinsics are required. Specifically, given the camera pose T_t and the foreground transformation F_t from our calibration (see Section 5.1), we define the camera extrinsics used by the baselines as

$$E_t = T_t^{-1} F_t. \tag{12}$$

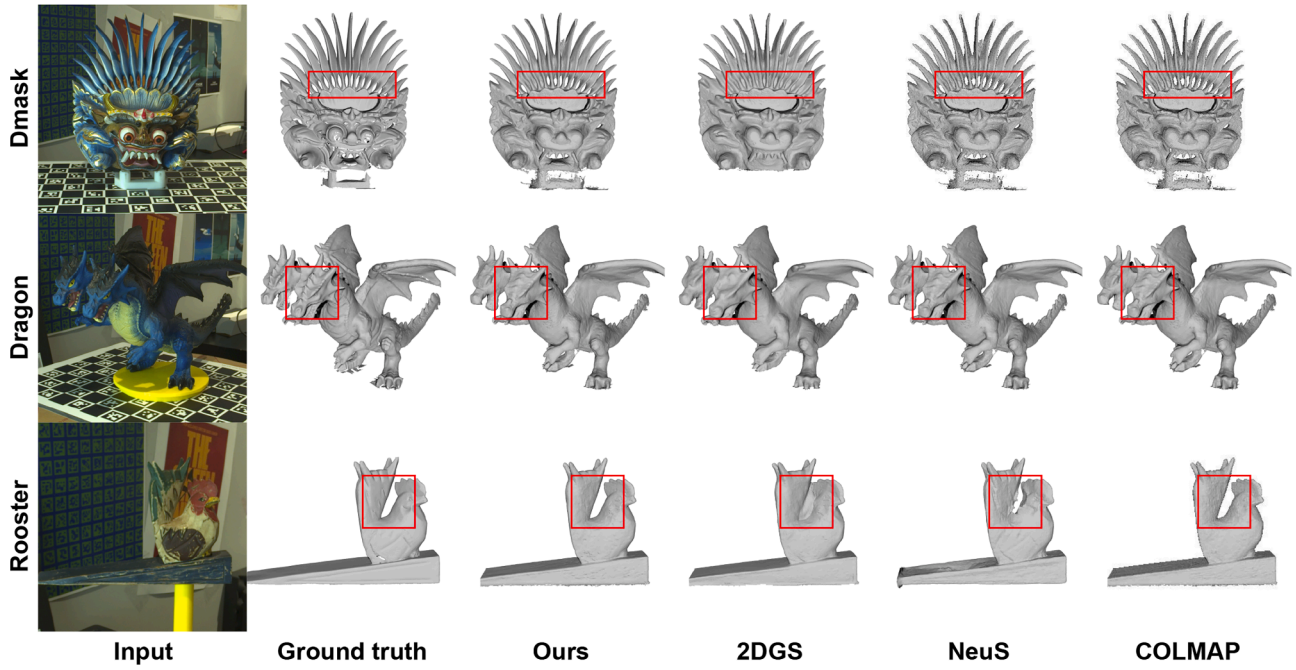


Fig. 5. Qualitative reconstruction results on the real dataset. Compared to other baselines, our method achieves finer-grained reconstructions, particularly in regions containing thin structures that are difficult to mask. This highlights the advantage of our mask-free separation approach.

For baselines that rely on provided foreground masks (i.e., do not explicitly model foreground/background separation), we use these calibrated extrinsics E_t, so that performance differences are not attributable to pose discrepancies.

5.3. Experimental results

A detailed comparison of baselines and our method on both real and synthetic datasets is conducted for 3D surface reconstruction and novel-view synthesis as presented in Tables 1–3. For all baselines, experimental results with and without masks on dynamic foreground objects are provided. To further demonstrate real-world applicability, we also evaluate our approach on two culturally significant wooden busts, highlighting its effectiveness for real world application.

5.3.1. 3D surface reconstruction

In Tables 1 and 2, we report the reconstruction quality in terms of Chamfer-$\mathcal{L}_1$ distances for all methods under both **w/** mask and **w/o** mask settings. For the synthetic dataset, we set a strong baseline 2DGS (GT) which we use ground truth masks with 2DGS. We observe that MFGS achieves performance comparable to this upper-bound setting, while still outperforming all other baselines by a clear margin. Remarkably, our mask-free method even surpasses 2DGS with segmentation masks from SAM [40]. We attribute this improvement to the fact that other methods rely on object masks for reconstruction, and imperfect masks can negatively affect their performance while our method learn to separate the foreground and is unaffected by mask quality.

Qualitative results in Fig. 5 reinforce these trends. For the *Dmask* object, neither NeuS nor 2DGS correctly preserves its sharp, outward spikes: NeuS over-smooths the tips, and 2DGS exaggerates their thick-

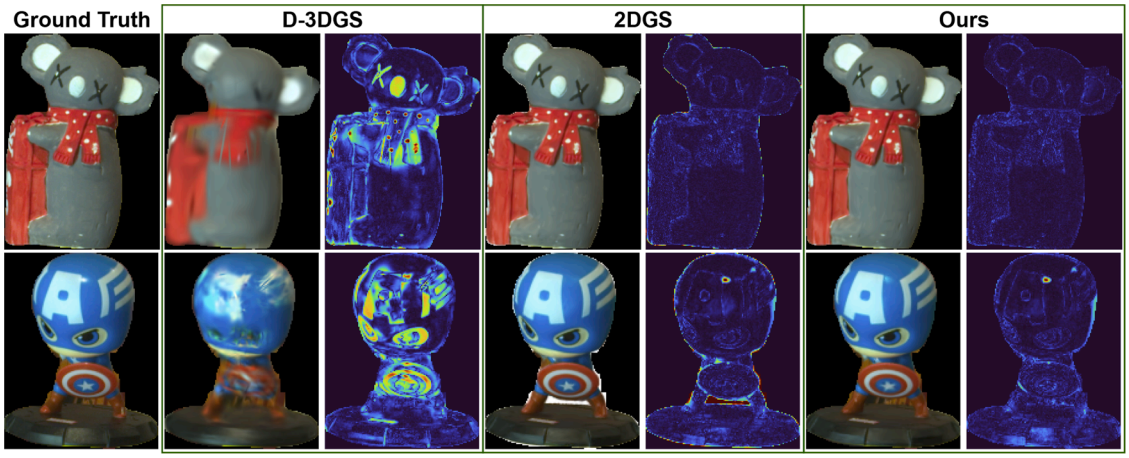


Fig. 6. Novel-view synthesis and corresponding error maps (MSE) on the real dataset. Our method outperforms D-3DGS by a large margin. Compared to 2DGS, our method exhibits significantly less error at the boundaries due to correctly modeling the background.

Table 2
Experiment results on the synthetic dataset. We report the Chamfer-$\mathcal{L}_1$ ↓ (scaled by 100×) and color each cell as best, second, and third. "–" indicates failure to reconstruct a mesh.

	Method	Crab	Insect	Leaves	Marci	Cockchafer	Miyuki	Pigeon	Plant1	Plant2	Avg.
mask	COLMAP	0.4393	0.3603	6.2447	1.1956	0.3907	0.8543	0.3992	2.1638	1.8153	1.5404
	NeuS	0.7346	0.2815	0.6058	0.6718	0.2689	0.7218	0.9289	2.4874	0.6879	0.8210
	2DGS (SAM)	0.4587	0.3551	0.6156	0.7757	0.3194	0.5548	0.5288	0.8849	0.8910	0.5982
	2DGS (GT)	0.4403	0.3532	0.5092	0.6210	0.3326	0.5516	0.5990	0.9335	0.5166	0.5397
w/o mask	NeuS	13.4557	2.7616	9.3365	30.1131	33.9812	29.8241	9.6987	13.4225	24.3946	18.5542
	2DGS	39.2561	43.8407	38.0376	89.3802	60.4679	53.3285	44.2398	-	31.5418	50.0116
	D-3DGS	13.2224	177.8401	68.6867	24.1406	22.0735	13.4585	63.2310	51.7127	31.3708	51.7485
	S2GS (ours)	0.4361	0.3297	0.5154	0.6481	0.3378	0.5592	0.5529	0.9829	0.5378	0.5444

ness. MFGS, however, retains the true geometry. A similar pattern appears on the *Rooster*: noisy masks cause 2DGS and NeuS to fuse background geometry into the gap between head and tail, whereas our method maintains a clean separation. We attribute this resilience to the two-level separation regularization applied in both 3D space and rendered 2D images. Together, these results demonstrate that MFGS delivers high-fidelity reconstructions without the fragile dependence on object masks.

5.3.2. *Novel-view synthesis*

As summarized in Table 3, MFGS delivers the best rendering quality across all metrics, outperforming every baseline by a considerable margin. Qualitative results in Fig. 6 corroborate these numbers: our renderings exhibit fewer artifacts, and the accompanying MSE maps show markedly lower error than those of D-3DGS in the mask-free setting. Even when NeuS and 2DGS are supplied with segmentation masks, MFGS still improves PSNR by more than on average. A closer look at the error maps reveals why: while interior regions match 2DGS in fidelity, our model produces far cleaner boundaries, thanks to its accurate, self-supervised separation of foreground and background.

5.3.3. *Analysis of Gaussian separation*

We assess segmentation quality on the synthetic dataset by comparing MFGS against 2DGS and SAM. For 2DGS we derive a mask from rendered opacity; for MFGS we use the confidence map introduced in Section 4.4. As shown in Table 6, Mask-IoU scores reveal that MFGS surpasses both 2DGS and SAM, confirming that our self-supervised separation remains reliable even under challenging conditions.

Qualitative results on real data are presented in Fig. 7. Although SAM generally produces plausible masks, it fails in difficult regions-e.g., the *Dmask* spikes and the gap between the *Rooster*'s head and tail-mis-segmentations that propagate directly into 2DGS reconstructions. By contrast, MFGS yields precise masks in the same regions, preventing foreground-background leakage and thereby boosting overall reconstruction quality. These findings highlight the advantage of our mask-free approach: it avoids the brittleness of external segmentation and

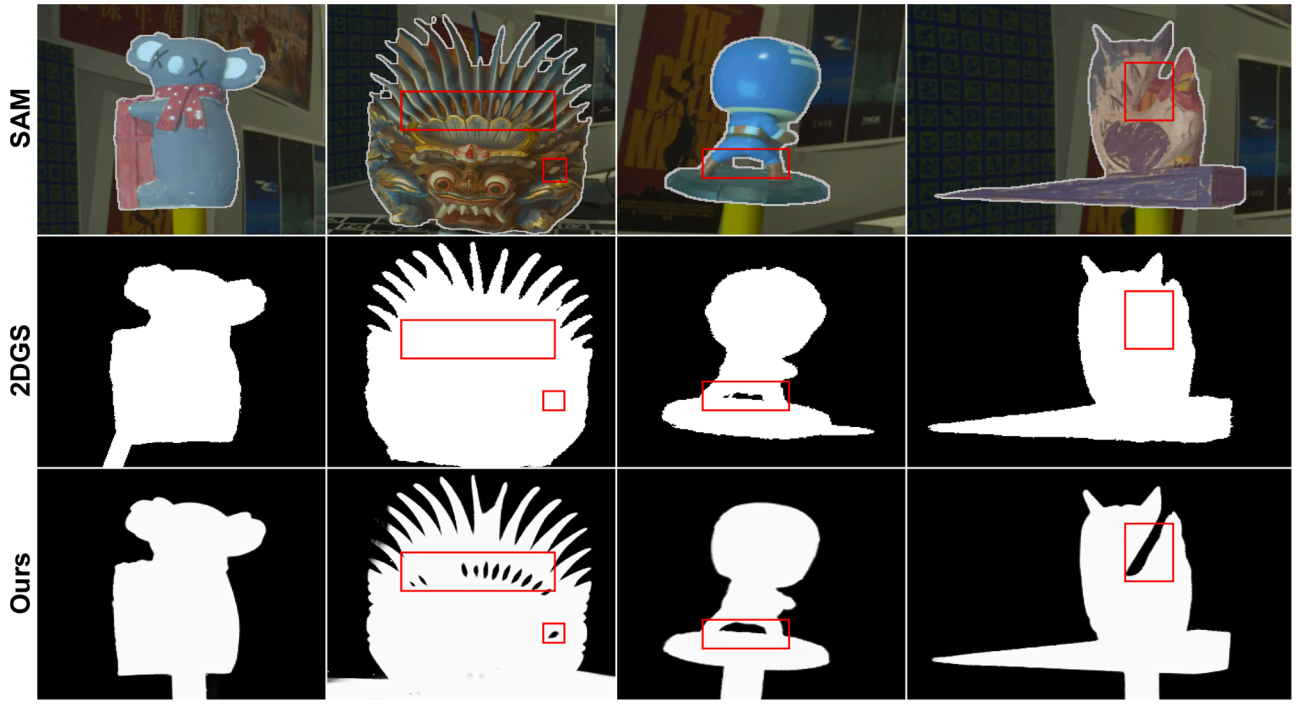


Fig. 7. Qualitative analysis of the segmentation masks. SAM achieves compelling segmentation masks generally, but it struggles in difficult regions highlighted. Those errors from SAM are further propagated to the reconstruction methods (2DGS). Our method achieves even better separation results by self-supervised optimization.

Table 3
Novel-view synthesis metrics on real and synthetic datasets.

	Method	Real dataset			Synthetic dataset		
		PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
w/ mask	NeuS	21.35	0.854	0.149	26.96	0.902	0.058
	2DGS	25.10	0.915	0.065	26.16	0.931	0.063
w/o mask	NeuS	18.36	0.456	0.458	19.41	0.648	0.280
	2DGS	12.00	0.469	0.637	15.64	0.693	0.471
	D-3DGS	23.42	0.807	0.201	19.76	0.787	0.201
	MFGS	31.86	0.946	0.047	33.89	0.969	0.037

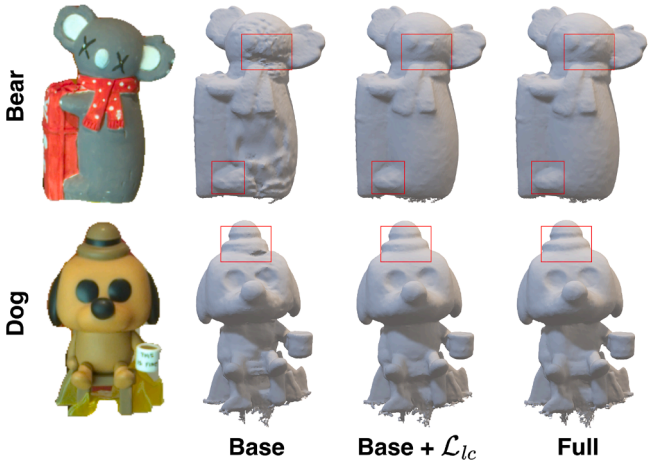


Fig. 8. Qualitative comparison of ablation variants. Adding the local consistency loss $\mathcal{L}_{lc}$ improves surface completeness and suppresses noisy Gaussians. Introducing the separation losses $\mathcal{L}_{2ds}$ and $\mathcal{L}_{3ds}$ further sharpens object boundaries and recovers fine details, producing the most accurate reconstruction.

Table 4
Analysis of the segmentation quality. We report the Mask-IoU on the synthetic dataset.

Method	2DGS	SAM	Ours
IoU	0.876	0.933	**0.977**

Table 5
Ablation study on the real dataset. Chamfer-$\mathcal{L}_1$ (↓) measures geometric accuracy, while PSNR (higher is better) measures novel-view synthesis quality. $\mathcal{L}_{lc}$ is the local consistency loss; $\mathcal{L}_s$ comprises the 3D and 2D separation regularizers.

Metrics	base	$+\mathcal{L}_{lc}$	$+\mathcal{L}_{lc} + \mathcal{L}_s$(full)
Chamfer-$\mathcal{L}_1$ ↓	0.1328	0.1296	0.1272
PSNR ↑	32.25	32.85	31.86

instead learns a consistent object/background split directly from multi-view supervision Table 4.

Table 5 summarizes how each regularization term progressively improves the base model on the real dataset, measured by Chamfer-$\mathcal{L}_1$ for geometry and PSNR for rendering.

5.4. Ablation study

5.4.1. Regularization terms

The corresponding qualitative comparisons in Fig. 8 further illustrate their effects on reconstruction quality. Introducing the local consistency loss $\mathcal{L}_{lc}$ produces clear improvements in both quantitative metrics and visual quality. Qualitatively, we observe that $\mathcal{L}_{lc}$ stabilizes the learned foreground probabilities and suppresses noisy or floating Gaussians, leading to smoother and more complete object surfaces. Adding the separation losses $\mathcal{L}_{2ds}$ and $\mathcal{L}_{3ds}$ further reduces the Chamfer distance error to its lowest level. The qualitative results show that these losses help refine object boundaries and recover fine details. Although PSNR drops marginally, reflecting the usual trade-off between strict geometric alignment and view-dependent appearance. Overall, both the numerical and visual comparisons demonstrate that each regularizer provides a meaningful improvement to geometry quality, while the full combination offers the highest surface accuracy, with only a minor compromise in photometric fidelity.

5.4.2. Analysis of hyperparameter sensitivity

We study the influence of the foreground probability threshold τ that classifies Gaussians as foreground or background. Fig. 9 shows that PSNR and Chamfer-$\mathcal{L}_1$ remain flat across a wide range of τ. This indicates that the method is insensitive to the exact choice of τ, and the gains are not due to a carefully tuned threshold.

5.4.3. Influence of mask quality

To quantify how baselines degrade with imperfect masks, we dilate the original SAM masks to produce nine quality levels; a larger index implies poorer quality, and ∞ denotes no mask. Results in Table 6 and Fig. 10 confirm that our method is agnostic to mask quality, whereas the baselines deteriorate sharply. NeuS fails on seven of nine objects when the quality index exceeds 11.

5.5. 2DGS with explicit SE(3) foreground transforms

To isolate the contribution of our learnable foreground probability p, we introduce a controlled baseline that augments standard 2DGS with the same calibrated per-frame SE(3) object transformations used in our method, but without learning any probabilistic foreground-background separation.

In this baseline-denoted 2DGS w/ pose, we begin by assigning Gaussian primitives to the foreground using a simple heuristic derived from dataset information (e.g., Gaussians whose centers fall within a predefined bounding cube are treated as foreground). These selected Gaussians are then explicitly transformed by the known per-frame SE(3) motion before rasterization, while all remaining Gaussians are fixed in the canonical background coordinate system. This setup corresponds to a “vanilla” 2DGS configuration equipped with

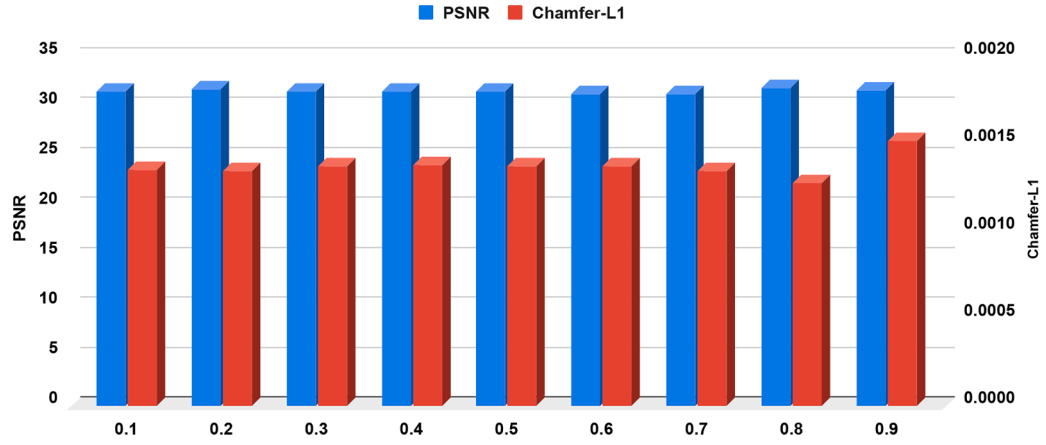


Fig. 9. Performance of our method vs. foreground probability threshold τ.

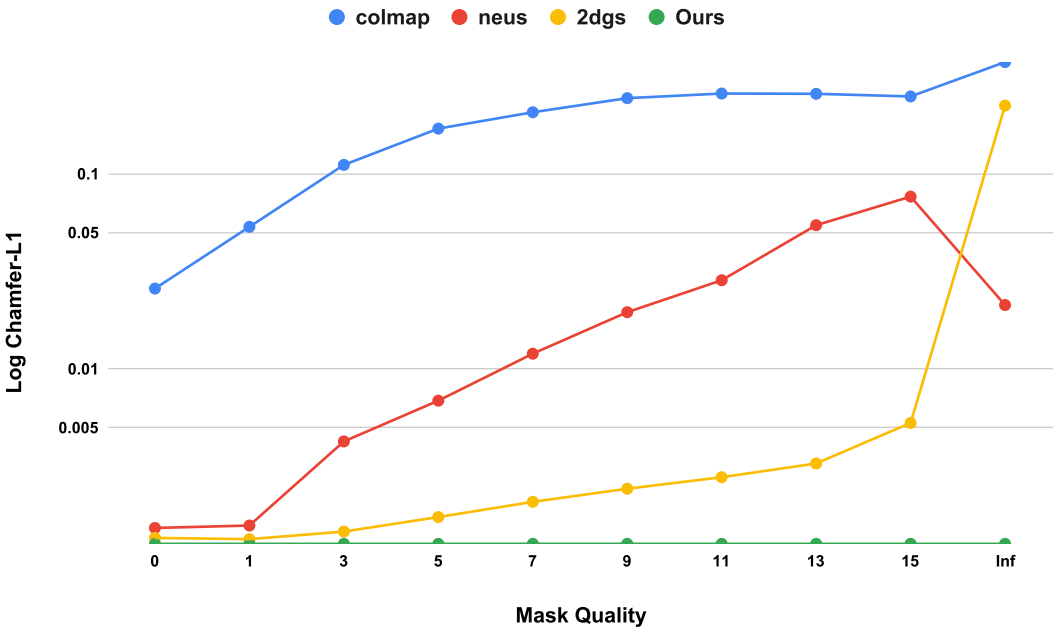


Fig. 10. Reconstruction metric (Chamfer-$\mathcal{L}_1$) w.r.t. different levels of mask quality. A higher quality value means a lower quality mask, and Inf means no mask is applied.

Table 6
Reconstruction evaluation metric Chamfer-$\mathcal{L}_1 \downarrow$ (scaled by 100×) on the real dataset with different mask quality. A higher quality value means a lower quality mask, and Inf means no mask is applied. The results show that the performance of other methods drops dramatically with worse image masks, while our method remains consistent since we don't rely on image masks.

Method	0	1	3	5	7	9	11	13	15	∞
COLMAP	2.5895	5.3659	11.1479	17.1102	20.7523	24.5067	25.8846	25.8010	25.0094	37.5152
NeuS	0.1535	0.1579	0.4261	0.6896	1.2004	1.9590	2.8565	5.4713	7.6668	2.1323
2DGS	0.1365	0.1345	0.1471	0.1745	0.2089	0.2439	0.2793	0.3286	0.5301	22.4533
Ours	**0.1272**	**0.1272**	**0.1272**	**0.1272**	**0.1272**	**0.1272**	**0.1272**	**0.1272**	**0.1272**	**0.1272**

the same pose information as our method but removes both the learnable probability p_i and the soft transformation mechanism described in Section 4.

Table 7 reports the performance of this baseline alongside 2DGS (SAM) and our full method. Across both synthetic and real datasets, 2DGS w/ pose does not match the accuracy of our approach, despite having access to the same calibrated SE(3) foreground motions. These results demonstrate that simply injecting pose information into 2DGS is insufficient; the key improvement stems from our probabilistic per-Gaussian labeling and differentiable soft transformation, which jointly enable the model to automatically discover the correct foreground/background split.

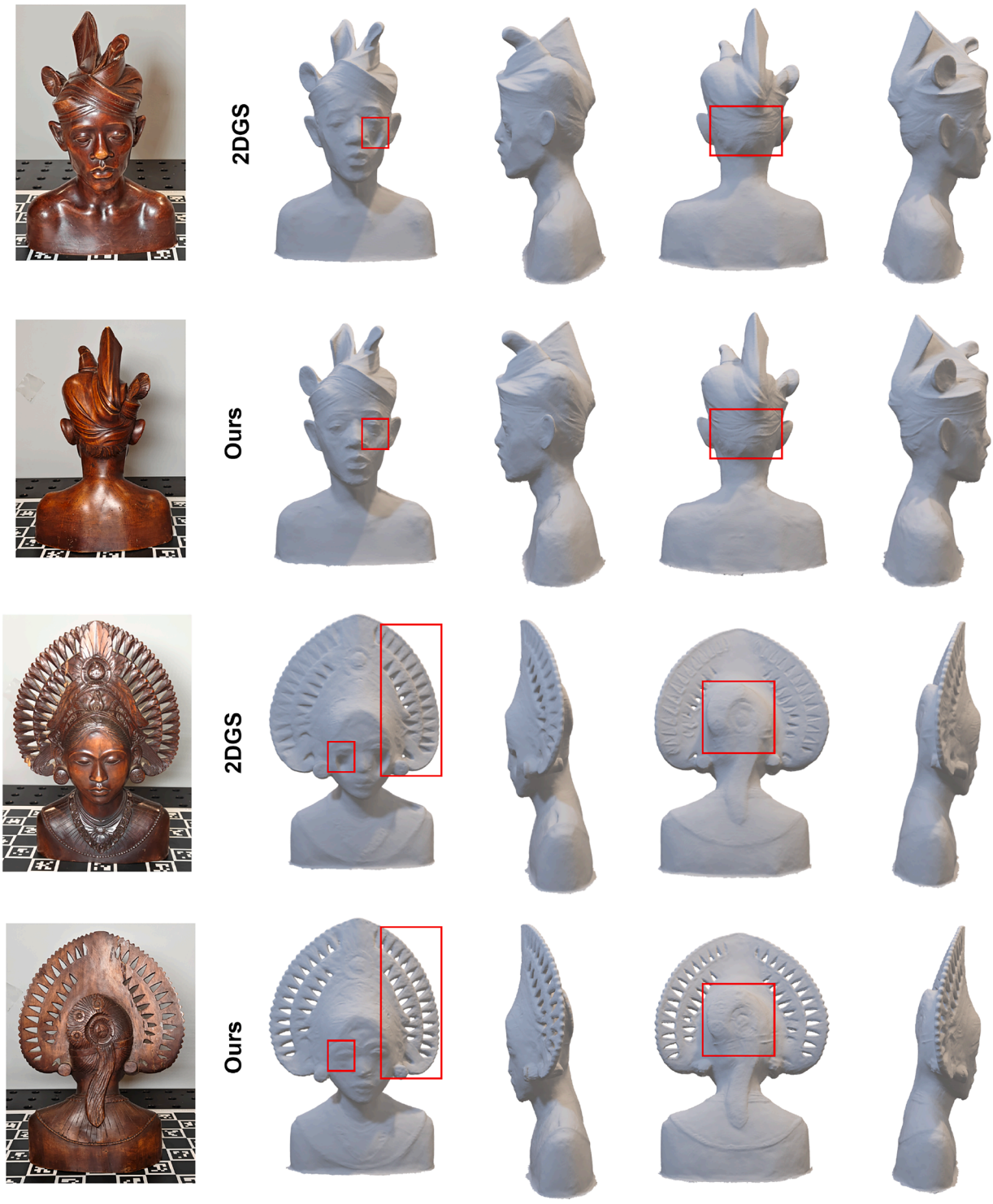


Fig. 11. Qualitative evaluation of our proposed method vs 2DGS on Balinese carved wooden sculptures: a young man (top) and a janger dancer (bottom). Compared to 2DGS with object masks from SAM, our mask-free approach demonstrates higher fidelity in reconstructing intricate details, such as surface holes and fine structures.

5.6. Experiments on cultural-heritage digitization

To demonstrate the practical value of our approach for cultural-heritage preservation, we scanned two wooden busts, *Young Man* and *Janger Dancer*, using the same turntable-robot setup. A short video of the capture process is provided in the supplementary material. These real artifacts present additional challenges: their wooden surfaces exhibit a mixture of subtle glossiness and large low-texture regions, while fine-scale carving such as hair strands, facial wrinkles, and headdress openings require precise geometric recovery.

As illustrated in Fig. 11, our mask-free pipeline reproduces fine geometric and textural details, capturing delicate facial contours, thin openings in the headdress, and even subtle surface wear, more consistently than 2DGS [15] aided by SAM masks [54]. While 2DGS often suffers from mask leakage or oversmoothing in low-texture regions, MFGS remains stable and preserves fine-grained structure.

Table 7
Chamfer-$\mathcal{L}_1$ (↓, scaled by 100×) on the real and synthetic datasets. 2DGS (SAM) uses SAM-predicted masks, 2DGS w/ pose applies the calibrated per-frame foreground transforms, and MFGS is our mask-free approach. We highlight the best results in **bold**.

Method	Real	Synthetic
2DGS (SAM)	0.1365	0.5982
2DGS w/ pose	0.1499	0.6102
MFGS (ours)	**0.1272**	**0.5444**

These results confirm that MFGS can generate high-fidelity digital replicas of cultural artifacts without any manual segmentation or preprocessing, highlighting its robustness, generalization ability, and suitability for real-world heritage digitization workflows where mask annotations are either noisy or infeasible to obtain.

6. Conclusion

We presented MFGS, a self-supervised framework that unifies 3D reconstruction and foreground-background separation without any external masks. By extending 2D Gaussian Splatting with learnable probabilistic labels and tailored regularizes, our method achieves state-of-the-art accuracy on both synthetic and real turntable datasets. It consistently surpasses classical SfM, NeRF-based, and Gaussian-splatting baselines, and even outperforms 2DGS equipped with high-quality SAM masks. Compared with the only other mask-free baseline, Deformable 3DGS, MFGS delivers markedly better geometry and rendering quality. Although our evaluation focuses on turntable scanning, the approach already proves useful in practical scenarios such as cultural-heritage digitization, where manual masking is often infeasible.

Limitations and future work. While promising, MFGS has several limitations. First, it assumes calibrated camera poses and an object-centric setup with a single rigid object rotating against a mostly static background. This restricts direct applicability to fully unconstrained, in-the-wild videos with hand-held cameras, complex backgrounds, or multiple independently moving objects. Second, our formulation is designed to separate one dominant foreground object from its background; scenes containing multiple interacting objects, strong occlusions, or layered foreground structures remain challenging and may require extending the model to support multiple foreground layers or hierarchical motion segmentation. Finally, highly non-rigid object motion or significant pose estimation errors can degrade the quality of separation and reconstruction.

In future work, we plan to generalize the self-supervised Gaussian-separation framework to fully dynamic scenes, multi-object environments, and multi-camera capture setups, further broadening its applicability to real-world 3D reconstruction scenarios.

CRediT authorship contribution statement

Jinguang Tong: Writing – review & editing, Writing – original draft, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **Xuesong Li:** Writing – review & editing, Methodology, Investigation, Formal analysis; **Sundaram Muthu:** Writing – original draft, Methodology, Data curation; **Fahira Afzal Maken:** Writing – review & editing, Methodology, Conceptualization; **Lars Petersson:** Writing – review & editing, Supervision, Conceptualization; **Chuong Nguyen:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization; **Hongdong Li:** Writing – review & editing, Supervision, Methodology.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Supplementary material

Supplementary material associated with this article can be found in the online version at 10.1016/j.patcog.2026.113341.

References

[1] R. Hartley, Multiple View Geometry in Computer Vision, 665, Cambridge university press, 2003.
[2] B. Mildenhall, P.P. Srinivasan, M. Tancik, J.T. Barron, R. Ramamoorthi, R. Ng, NeRF: representing scenes as neural radiance fields for view synthesis, Commun. ACM 65 (1) (2021) 99–106.
[3] Q. Fu, Q. Xu, Y.S. Ong, W. Tao, Geo-Neus: geometry-consistent neural implicit surfaces learning for multi-view reconstruction, Adv. Neural Inf. Process. Syst. 35 (2022) 3403–3416.
[4] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, W. Wang, NeuS: learning neural implicit surfaces by volume rendering for multi-view reconstruction, Adv. Neural Inf. Process. Syst. 34 (2021) 27171–27183.
[5] Y. Wang, Q. Han, M. Habermann, K. Daniilidis, C. Theobalt, L. Liu, Neus2: fast learning of neural implicit surfaces for multi-view reconstruction, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 3295–3306.
[6] J. Tong, S. Muthu, F.A. Maken, C. Nguyen, H. Li, Seeing through the glass: neural 3D reconstruction of object inside a transparent container, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 12555–12564.
[7] B. Kerbl, G. Kopanas, T. Leimkühler, G. Drettakis, 3D Gaussian splatting for real-time radiance field rendering, ACM Trans. Graph. 42 (4) (2023) 139–1.
[8] A. Guédon, V. Lepetit, SuGaR: surface-aligned Gaussian splatting for efficient 3D mesh reconstruction and high-quality mesh rendering, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024, pp. 5354–5363.
[9] P. Dai, J. Xu, W. Xie, X. Liu, H. Wang, W. Xu, High-quality surface reconstruction using Gaussian surfels, in: ACM SIGGRAPH 2024 Conference Papers, 2024, pp. 1–11.
[10] J. Tong, X. Li, F.A. Maken, S. Muthu, L. Petersson, C. Nguyen, H. Li, GS-2DGS: geometrically supervised 2DGS for reflective object reconstruction, in: Proceedings of the Computer Vision and Pattern Recognition Conference, 2025, pp. 21547–21557.
[11] C. Zhao, X. Huang, K. Yang, X. Wang, Q. Wang, Generalizable 3D Gaussian splatting for novel view synthesis, Pattern Recognit. 161 (2025) 111271.
[12] C. Media, 3D-twin scanner, 2024, https://www.covisionmedia.ai/en.
[13] Z. Dong, K. Chen, Z. Lv, H.-X. Yu, Y. Zhang, C. Zhang, Y. Zhu, S. Tian, Z. Li, G. Moffatt, et al., Digital twin catalog: a large-scale photorealistic 3d object digital twin dataset, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2025, pp. 753–763.
[14] I.D. Lee, J.H. Seo, Y.M. Kim, J. Choi, S. Han, B. Yoo, Automatic pose generation for robotic 3-D scanning of mechanical parts, IEEE Trans. Rob. 36 (4) (2020) 1219–1238.
[15] B. Huang, Z. Yu, A. Chen, A. Geiger, S. Gao, 2D Gaussian splatting for geometrically accurate radiance fields, in: ACM SIGGRAPH 2024 Conference Papers, 2024, pp. 1–11.
[16] L. Ke, M. Ye, M. Danelljan, Y.-W. Tai, C.-K. Tang, F. Yu, et al., Segment anything in high quality, Adv. Neural Inf. Process. Syst. 36 (2023) 29914–29934.
[17] L. Yariv, Y. Kasten, D. Moran, M. Galun, M. Atzmon, B. Ronen, Y. Lipman, Multiview neural surface reconstruction by disentangling geometry and appearance, Adv. Neural Inf. Process. Syst. 33 (2020) 2492–2502.
[18] M. Niemeyer, L. Mescheder, M. Oechsle, A. Geiger, Differentiable volumetric rendering: learning implicit 3D representations without 3D supervision, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3504–3515.
[19] M. Oechsle, S. Peng, A. Geiger, UNISURF: unifying neural implicit surfaces and radiance fields for multi-view reconstruction, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 5589–5599.
[20] L. Yariv, J. Gu, Y. Kasten, Y. Lipman, Volume rendering of neural implicit surfaces, Adv. Neural Inf. Process. Syst. 34 (2021) 4805–4815.
[21] Z. Yu, S. Peng, M. Niemeyer, T. Sattler, A. Geiger, MonoSDF: exploring monocular geometric cues for neural implicit surface reconstruction, Adv. Neural Inf. Process. Syst. 35 (2022) 25018–25032.
[22] Z. Li, T. Müller, A. Evans, R.H. Taylor, M. Unberath, M.-Y. Liu, C.-H. Lin, Neuralangelo: high-fidelity neural surface reconstruction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 8456–8465.
[23] T. Müller, A. Evans, C. Schied, A. Keller, Instant neural graphics primitives with a multiresolution hash encoding, ACM Trans. Graph. (TOG) 41 (4) (2022) 1–15.

[24] Y. Wang, X. He, S. Peng, H. Lin, H. Bao, X. Zhou, AutoRecon: automated 3D object discovery and reconstruction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 21382–21391.
[25] H. Bai, Y. Chen, L. Wang, High-fidelity mask-free neural surface reconstruction for virtual reality, arXiv: 2409.13158 (2024).
[26] H. Kong, X. Yang, X. Wang, Generative sparse-view Gaussian splatting, in: Proceedings of the Computer Vision and Pattern Recognition Conference, 2025, pp. 26745–26755.
[27] X. Li, J. Tong, J. Hong, V. Rolland, L. Petersson, DGNS: deformable Gaussian splatting and dynamic neural surface for monocular dynamic 3D reconstruction, in: Proceedings of the 33rd ACM International Conference on Multimedia, 2025, pp. 1812–1821.
[28] Z. Jiang, H. Rahmani, S. Black, B. Williams, 3D points splatting for real-time dynamic hand reconstruction, Pattern Recognit. (2025) 111426.
[29] J. Yin, W. Yin, H. Chen, X. Ren, Z. Ma, J. Guo, Y. Liu, HumanRecon: neural reconstruction of dynamic human using geometric cues and physical priors, Pattern Recognit. (2025) 111964.
[30] H. Chen, C. Li, G.H. Lee, NeuSG: Neural implicit surface reconstruction with 3D Gaussian splatting guidance, arXiv: 2312.00846 (2023).
[31] X. Lyu, Y.-T. Sun, Y.-H. Huang, X. Wu, Z. Yang, Y. Chen, J. Pang, X. Qi, 3DGSR: implicit surface reconstruction with 3D Gaussian splatting, ACM Trans. Graph. (TOG) 43 (6) (2024) 1–12.
[32] J. Luiten, G. Kopanas, B. Leibe, D. Ramanan, Dynamic 3D Gaussians: tracking by persistent dynamic view synthesis, in: 2024 International Conference on 3D Vision (3DV), IEEE, 2024, pp. 800–809.
[33] Z. Yang, X. Gao, W. Zhou, S. Jiao, Y. Zhang, X. Jin, Deformable 3D Gaussians for high-fidelity monocular dynamic scene reconstruction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 20331–20341.
[34] Z. Guo, W. Zhou, L. Li, M. Wang, H. Li, Motion-aware 3D Gaussian splatting for efficient dynamic scene reconstruction, IEEE Trans. Circuits Syst. Video Technol. 35 (4) (2024) 3119–3133.
[35] S. Wang, X. Yang, Q. Shen, Z. Jiang, X. Wang, Gflow: recovering 4D world from monocular video, in: Proceedings of the AAAI Conference on Artificial Intelligence, 39, 2025, pp. 7862–7870.
[36] G. Wu, T. Yi, J. Fang, L. Xie, X. Zhang, W. Wei, W. Liu, Q. Tian, X. Wang, 4D Gaussian splatting for real-time dynamic scene rendering, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 20310–20320.
[37] Z. Yang, H. Yang, Z. Pan, L. Zhang, Real-time photorealistic dynamic scene representation and rendering with 4D Gaussian splatting, in: International Conference on Learning Representations (ICLR), 2024.
[38] Z. Lu, J. Ye, J. Leonard, 3DGS-CD: 3D Gaussian splatting-based change detection for physical object rearrangement, IEEE Rob. Autom. Lett. 10 (3) (2025) 2662–2669.
[39] Y. Xiong, B. Varadarajan, L. Wu, X. Xiang, F. Xiao, C. Zhu, X. Dai, D. Wang, F. Sun, F. Iandola, et al., EfficientSAM: leveraged masked image pretraining for efficient segment anything, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 16111–16121.
[40] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A.C. Berg, W.-Y. Lo, et al., Segment anything, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 4015–4026.
[41] M. Ye, M. Danelljan, F. Yu, L. Ke, Gaussian grouping: segment and edit anything in 3D scenes, in: European Conference on Computer Vision, Springer, 2024, pp. 162–179.
[42] J. Cen, J. Fang, C. Yang, L. Xie, X. Zhang, W. Shen, Q. Tian, Segment any 3D Gaussians, in: Proceedings of the AAAI Conference on Artificial Intelligence, 39, 2025, pp. 1971–1979.
[43] A.A. Kirillov, An Introduction to Lie Groups and Lie Algebras, 113, Cambridge University Press, 2008.
[44] J.L. Schönberger, J.-M. Frahm, Structure-from-motion revisited, in: Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
[45] X. Zou, J. Yang, H. Zhang, F. Li, L. Li, J. Wang, L. Wang, J. Gao, Y.J. Lee, Segment everything everywhere all at once, Adv. Neural Inf. Process. Syst. 36 (2023), 19769–19782.
[46] A. Chen, Z. Xu, A. Geiger, J. Yu, H. Su, TensoRF: tensorial radiance fields, in: European Conference on Computer Vision, Springer, 2022, pp. 333–350.
[47] R. Hartley, J. Trumpf, Y. Dai, H. Li, Rotation averaging, Int. J. Comput. Vis. 103 (2013) 267–305.
[48] E. Wigner, Group theory: and its application to the quantum mechanics of atomic spectra, Elsevier, 2012.
[49] R.A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A.J. Davison, P. Kohi, J. Shotton, S. Hodges, A. Fitzgibbon, Kinectfusion: real-time dense surface mapping and tracking, in: 2011 10th IEEE International Symposium on Mixed and Augmented Reality, Ieee, 2011, pp. 127–136.
[50] Q.-Y. Zhou, J. Park, V. Koltun, Open3D: a modern library for 3D data processing, arXiv: 1801.09847 (2018).
[51] B.O. Community, Blender - a 3D modelling and rendering package, Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. http://www.blender.org.
[52] G. Guidi, B.D. Frischer, 3D digitization of cultural heritage, in: 3D imaging, analysis and applications, 631–697, Springer, 2020.
[53] Y. Hou, S. Kenderdine, D. Picca, M. Egloff, A. Adamou, Digitizing intangible cultural heritage embodied: state of the art, J. Comput. Cult. Heritage 15 (3) (2022) 1–20.
[54] J. Yang, M. Gao, Z. Li, S. Gao, F. Wang, F. Zheng, Track anything: segment anything meets videos, arXiv: 2304.11968 (2023).