

6DOPE-GS: Online 6D Object Pose Estimation using Gaussian Splatting

Yufeng Jin^{1,2}, Vignesh Prasad¹, Snehal Jauhri¹, Mathias Franzius², Georgia Chalvatzaki^{1,3}

¹Computer Science Department, Technische Universität Darmstadt, Germany

²Honda Research Institute Europe GmbH, Offenbach, Germany ³Hessian.AI, Darmstadt, Germany

{yufeng.jin, vignesh.prasad, snehal.jauhri}@tu-darmstadt.de,
 georgia.chalvatzaki@tu-darmstadt.de

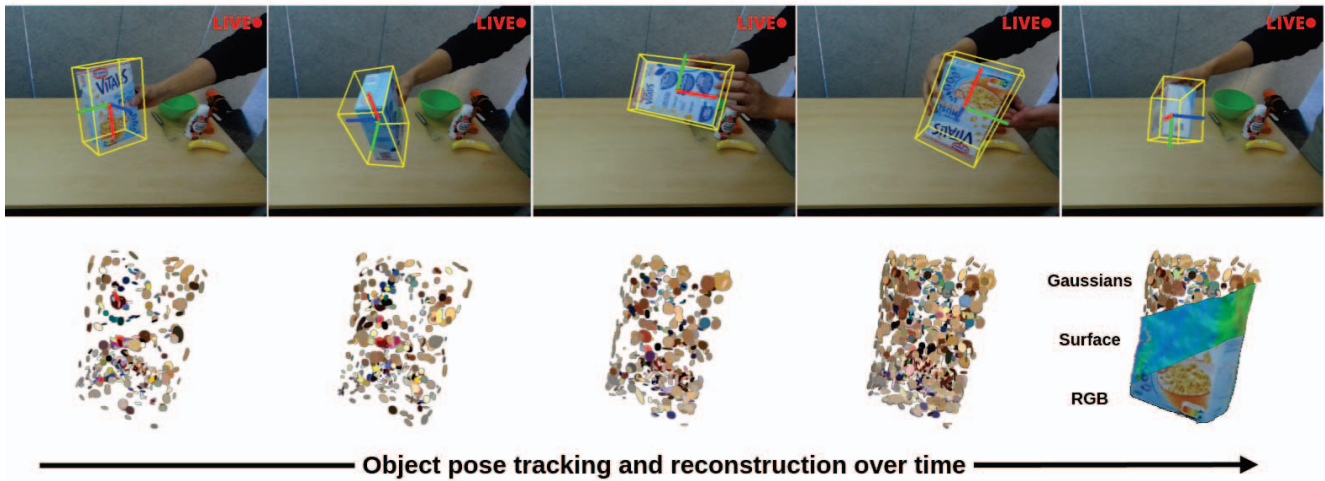


Figure 1. Demonstrating live object pose tracking and reconstruction of a test object in the real-world using 6DOPE-GS: a novel method for joint 6D object pose estimation and reconstruction using Gaussian Splatting. **Top:** 6D pose estimates of the object over time, **Bottom:** Example reconstruction over time with 2D Gaussian disks used to render the surface and appearance of the object. Our method enables live pose tracking and Gaussian Splat reconstruction of dynamic objects at 3.5Hz.

Abstract

Efficient and accurate object pose estimation is an essential component for modern vision systems in many applications such as Augmented Reality, autonomous driving, and robotics. While research in model-based 6D object pose estimation has delivered promising results, model-free methods are hindered by the high computational load in rendering and inferring consistent poses of arbitrary objects in a live RGB-D video stream. To address this issue, we present 6DOPE-GS, a novel method for online 6D object pose estimation and tracking with a single RGB-D camera by effectively leveraging advances in Gaussian Splatting. Thanks to the fast differentiable rendering capabilities of Gaussian Splatting, 6DOPE-GS can simultaneously optimize for 6D object poses and 3D object reconstruction. To achieve the necessary efficiency and accuracy for live tracking, our method uses incremental 2D Gaussian Splatting with an intelligent dynamic keyframe selection procedure to

achieve high spatial object coverage and prevent erroneous pose updates. We also propose an opacity statistic-based pruning mechanism for adaptive Gaussian density control, to ensure training stability and efficiency. We evaluate our method on the HO3D and YCBInEOAT datasets and show that 6DOPE-GS matches the performance of state-of-the-art baselines for model-free simultaneous 6D pose tracking and reconstruction while providing a 5× speedup. We also demonstrate the method’s suitability for live, dynamic object tracking and reconstruction in a real-world setting.

1. Introduction

Precise tracking and accurate reconstruction of objects allows capturing essential spatial and structural information, essential for downstream tasks such as robotic manipulation [10, 33, 61], augmented reality [62, 70, 83], automation [15, 28], assisted robot teleoperation [39], etc. The majority of 6D object pose estimation and tracking methods,

whether for seen or unseen objects, have primarily used model-based techniques. Several approaches [30, 31, 34, 46, 78, 85] use CAD models rendered from various angles during training and perform feature matching at inference time for rapid pose estimation. Such model-based approaches augmented with synthetic training data [78] have shown state-of-the-art performance instance-level pose estimation. However, doing so requires either a CAD model or a small set of reference images annotated with the object poses, which becomes tedious as the number of unseen objects increases.

On the other hand, there has been exciting progress in zero-shot, model-free methods over the past few years [75, 77] which require no additional prior information, other than an object mask. BundleSDF [77] operates in a model-free manner by jointly optimizing a “neural object field” and the object poses by learning a 3D Signed Distance Field representation while concurrently running a global pose graph optimization. However, despite reporting near real-time pose optimization capabilities ($\sim 10\text{Hz}$), the neural object field training is far from real-time¹, which limits the average tracking frequency to $\sim 0.4\text{Hz}$. The significant computational overhead associated with training the neural object field hinders its applicability in live dynamic scenarios, where rapid pose updates are crucial.

To address this limitation, we leverage Gaussian Splatting [23, 27] which offers significantly better computational efficiency for real-time applications. We propose a novel method for online 6D object pose estimation through Gaussian Splatting, “6DOPE-GS”, that enables model-free, live object tracking and reconstruction. Building upon recent advances in using Gaussian Splatting for SLAM [26], 6DOPE-GS jointly optimizes object poses from observed keyframes and reconstructs a 3D object model on the fly using incremental 2D Gaussian Splatting [23]. We propose several algorithmic enhancements to attain the required accuracy, efficiency, and training stability for live reconstruction and tracking. For accuracy, our method uses a novel dynamic keyframe selection mechanism to prioritize spatial coverage of the object and reconstruction confidence-based filtering to exclude keyframes with erroneous pose estimates. To maintain training stability and efficiency, we propose an adaptive Gaussian density control mechanism based on the opacity statistics of the Gaussians. Our contributions provide a significant speed-up in object pose estimation and tracking while maintaining high accuracy. In particular, we evaluate 6DOPE-GS on the HO3D and YCBInEOAT datasets and observe that it matches the state-of-the-art performance of competitive baselines while providing a $5\times$ speedup. We also demonstrate the live, dynamic object tracking and reconstruction ability of 6DOPE-GS in a real-world setting. To the best of our knowledge, we are the first method to perform joint object tracking and Gaussian Splat reconstruction live at 3.5Hz

¹<https://github.com/NVlabs/BundleSDF/issues/122>

from a single RGB-D camera.

Our contributions are as follows:

- We propose a novel method that effectively leverages 2D Gaussian Splatting for efficient and accurate model-free 6D object pose estimation and reconstruction.
- We leverage the computationally efficient differentiable rendering of Gaussian Splatting to jointly optimize a 2D Gaussian Splatting-based “Gaussian Object Field” along with an object-centric pose graph of observed keyframes, that provides accurate, refined keyframe pose updates.
- We propose a dynamic keyframe selection approach based on the spatial coverage of the set of keyframes and a reconstruction confidence-based filtering mechanism to exclude keyframes with erroneous pose estimates.
- We incorporate a novel adaptive Gaussian density control mechanism based on opacity percentiles to filter out “unimportant” Gaussians, thereby improving training stability and computational efficiency.

2. Related Work

2.1. Object Pose Estimation and Tracking

Instance-level 6D object pose estimation typically requires object CAD models and/or pretraining [4, 19, 20, 29, 34, 53, 68, 69, 72, 76]. Such instance-level methods can be further categorized into correspondence-based [52, 54, 67], template-based [9, 65], voting-based [19, 20, 35, 46, 69], and regression-based [14, 22] methods. For better generalization, some approaches use an object CAD model only at inference time [30, 59, 85]. Other methods [3, 18, 21, 36, 50, 64, 78] relax this assumption by utilizing only few reference images of the object instead of the CAD model.

BundleTrack [75] enables near real-time ($\sim 10\text{Hz}$), model-free tracking with a SLAM-style approach. It uses keyframe point correspondences for coarse pose initialization with RANSAC, followed by object-centric pose graph optimization for refined estimates. BundleSDF [77] extends this by jointly performing pose tracking and object reconstruction through a neural object field, achieving state-of-the-art results in model-free settings. However, the neural field training is slow and computationally demanding ($\sim 6.7\text{s}$ per training round [77]), limiting its real-time applicability. We address this key limitation by leveraging the efficiency of Gaussian Splatting for joint object reconstruction and pose refinement, enabling effective live tracking.

2.2. 3D Reconstruction

3D Reconstruction is a well-studied problem in Photogrammetry. Structure from Motion (SfM) [48] is a commonly used approach to estimate camera poses and a sparse 3D structure from images without prior pose knowledge in an offline manner. Multi-View Stereo approaches (MVS) [13, 74] build upon such pose estimates to refine a dense 3D re-

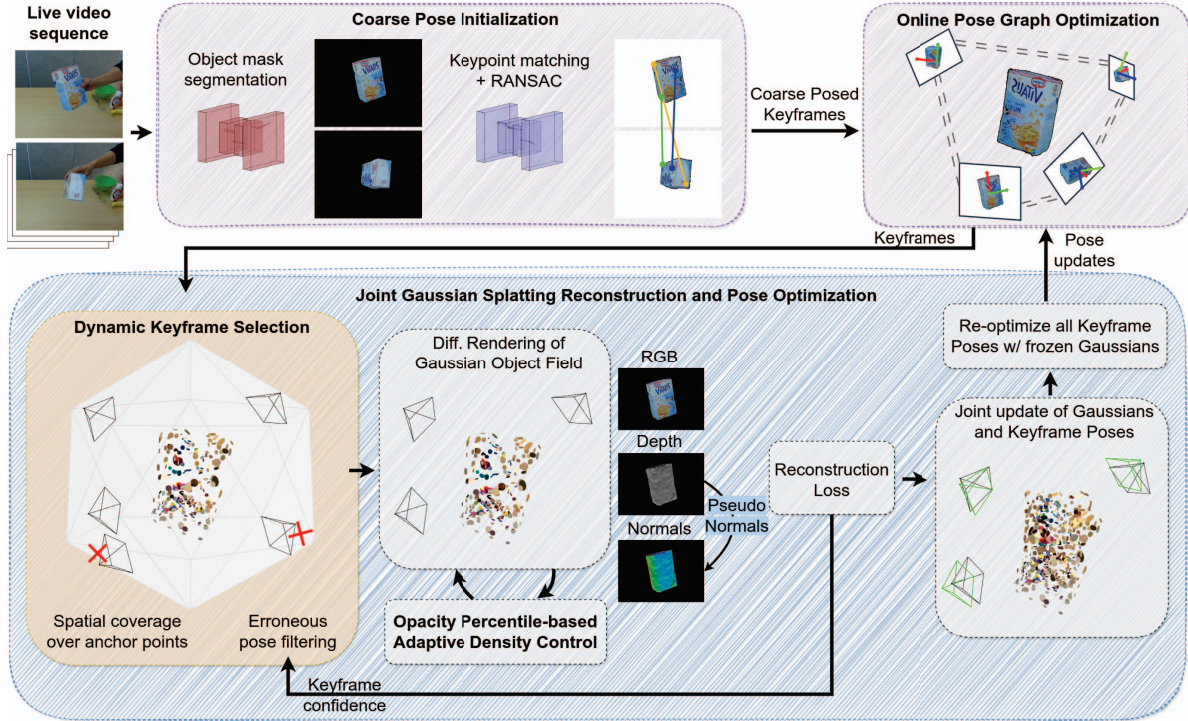


Figure 2. Overview of our approach: 6DOPE-GS. Given a live input RGB-D video stream, we obtain object segmentation masks using SAM2 [55] on the incoming video frames. We then use LoFTR [63], a transformer-based feature matching approach, to obtain pairwise correspondences between multiple views. We initialize a set of “keyframes” based on the density of matched features, for which we establish initial coarse pose estimates using RANSAC. To obtain refined pose updates for the keyframes, we use a 2D Gaussian Splatting-based “Gaussian Object Field” that is jointly optimized with the keyframe poses in a concurrent thread. We filter out erroneous keyframes for accurate pose refinement updates using a novel dynamic keyframe selection mechanism based on spatial coverage and reconstruction confidence. Moreover, we incorporate an opacity percentile-based adaptive density control mechanism to prune out inconsequential Gaussians, thus improving training stability and efficiency. Once the Gaussian Object Field is updated, it is temporarily frozen and the poses of keyframes that were filtered out are also updated. The object pose estimate at each timestep is then obtained by performing an online pose graph optimization using the incoming keyframe with the current set of keyframes.

construction. For enabling more real-time reconstruction and pose tracking, Simultaneous Localization and Mapping (SLAM) methods [5, 25, 38] approach the problem by jointly optimizing the camera poses and the environment reconstruction. Emerging methods that leverage neural representations have enhanced the fidelity of 3D reconstructions [56, 71, 73, 82]. Along similar lines, the use of Neural Radiance Fields (NeRFs) [43] and Signed Distance Fields (SDFs) [7, 42, 47, 49], with their volumetric rendering approach provide highly photorealistic reconstructions.

Gaussian Splatting [27] is a particle-based alternative that models scene density with Gaussian distributions and achieves significantly faster rendering speeds with similar levels of photorealism by using rasterization of explicit Gaussian particles, thereby avoiding the ray-marching steps used by volumetric rendering methods. Recently, 2D Gaussian Splatting [23] has improved the surface rendering capabilities of Gaussian Splatting by optimizing oriented planar 2D Gaussian disks close to the scene surfaces. However, all these

methods still depend on pre-established camera poses. Coming from a SLAM perspective, recent approaches explore jointly optimizing camera pose estimates as well as the map reconstruction that uses Gaussian Splatting [26, 37, 41, 81]. In this work, we propose a novel method that extends scene-level approaches to object-level tracking and reconstruction by leveraging the SLAM-inspired capabilities for object tracking [75, 77] and Gaussian Splatting [26, 41, 81] with the precise surface rendering capabilities offered by 2D Gaussian Splatting [23].

3. Method

We introduce a novel method for real-time 6D Object Pose Estimation using the representation capabilities of 2D Gaussian Splatting. Fig. 2 presents a schematic overview of our approach. To accurately track the 6DoF pose of an object captured by a single RGB-D camera, we start by segmenting the object in the first frame using SAM2 [55] to ensure

precise object segmentation throughout the video sequence. With the object segmented across multiple frames, we apply LoFTR [63] to establish point correspondences, identifying keyframes for a Coarse Pose Initialization via Bundle Adjustment [75] (Sec. 3.1). This initial set of coarsely estimated keyframes is then refined through a joint optimization with 2D Gaussians using differentiable rendering, yielding accurate pose corrections and an improved object model for the keyframes (Sec. 3.2). To improve the quality of the generated 3D model and to subsequently enable a more accurate pose refinement, we propose a dynamic keyframe selection technique for selecting the best keyframes for optimizing the 2D Gaussians based on their estimated spatial coverage around the object and their reconstruction accuracy (Sec. 3.3). During this phase, we iteratively employ a novel pruning/adaptive density control mechanism to stabilize the number of Gaussian particles required, to balance computational efficiency with reconstruction accuracy (Sec. 3.4). Once the joint optimization converges, all the keyframe poses are subsequently optimized and help guide the Online Pose Graph Optimization (Sec. 3.5) in continuously refining the object pose at each subsequent timestep for robust and precise tracking.

3.1. Coarse Pose Initialization

To enable real-time 6D pose tracking and reconstruction of arbitrary objects, we first use SAM2 [55] for facilitating effective segmentation and tracking of the object in question. Specifically, we use a fixed-length window of past frames combined with prompted images as input. We then use LoFTR [63], a transformer-based dense feature matcher, to estimate feature point correspondences between neighboring images. Using these matches, we compute a coarse pose estimate between pairs of RGB-D frames with nonlinear least-squares optimization [1] in a RANSAC fashion [12]. Subsequently, a keyframe memory pool is created wherein if an incoming frame is deemed to be spatially diverse compared to the existing pool, it is added as a new keyframe. Further details regarding the feature matching and the keyframe memory pool initialization are in [77].

3.2. Gaussian Object Field

To build an internal model that captures the visual and geometric properties of the object in an efficient and accurate manner, we build a Gaussian Object Field using 2D Gaussian Splatting (2DGS) [23] to achieve precise surface geometry reconstruction. Unlike 3D Gaussian Splatting (3DGS) [27], which primarily emphasizes on rendering realistic visual effects, 2DGS ensures accurate geometric alignment by converting each Gaussian into a disk-like surfel. This surfel-based approach, combined with our novel dynamic keyframe selection (Sec. 3.3) and opacity quartile-based pruning (Sec. 3.4), enables 2DGS to precisely model

the rendered surface, thereby delivering reliable depth estimates and addressing the limitations observed in 3DGS.

In 3DGS, a scene is represented as a set of 3D Gaussian particles, each of which represents a 3D distribution and is defined by its 3D centroid (mean) $\boldsymbol{\mu} \in \mathbb{R}^3$ and a covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{3 \times 3}$ which can be decomposed into a diagonalized scaling matrix $\boldsymbol{S} = \text{diag}([s_x, s_y, s_z])$ and a rotation matrix $\boldsymbol{R} \in SO(3)$ as $\boldsymbol{\Sigma} = \boldsymbol{R}\boldsymbol{S}\boldsymbol{S}^\top\boldsymbol{R}^\top$, which denotes the volume (spread) of the particle in 3D space. Along with the mean and covariance, each Gaussian is further characterized by spherical harmonic coefficients $c \in \mathbb{R}^k$ to represent view-dependent appearance, and an opacity value $\alpha \in [0, 1]$. For rendering, each 3D Gaussian is converted to camera coordinates using the world-to-camera transformation matrix \boldsymbol{W} and mapped to the image plane via a local affine transformation \boldsymbol{J} , $\boldsymbol{\Sigma}' = \boldsymbol{J}\boldsymbol{W}\boldsymbol{\Sigma}\boldsymbol{W}^\top\boldsymbol{J}^\top$. Once the 3D Gaussian is “splatted” onto the image plane, excluding the third row and column of $\boldsymbol{\Sigma}'$ results in a 2D covariance matrix $\boldsymbol{\Sigma}^{2D}$ that represents a 2D Gaussian G^{2D} in the image plane. The Gaussians are first ordered in ascending order based on their distance to the camera origin. Using volumetric rendering, we calculate the per-pixel color estimates $\hat{c}(\boldsymbol{p})$ of a pixel $\boldsymbol{p} = [u, v]^\top$ as the α -blending of N ordered Gaussians from front to back along the view direction

$$\hat{c}(\boldsymbol{p}) = \sum_{i \in N} c_i \alpha_i G_i^{2D}(\boldsymbol{p}) \prod_{j=1}^{i-1} (1 - \alpha_j G_j^{2D}(\boldsymbol{p})), \quad (1)$$

where α_i and c_i denote the opacity and the view-dependent appearance of the i th Gaussian, respectively. The depth image can be similarly rendered by replacing c_i with the z -depth coordinate of the i th Gaussian in the camera frame.

For 2DGS [23], the z -component of the scaling matrix is set to zero $\boldsymbol{S} = \text{diag}([s_u, s_v, 0])$ for each Gaussian, thereby collapsing the 3D volume into a set of 2D oriented planar Gaussian disks with two principal axes \boldsymbol{t}_u and \boldsymbol{t}_v . The normal to the 2D Gaussian can then be defined as $\boldsymbol{t}_w = \boldsymbol{t}_u \times \boldsymbol{t}_v$, which allows us to define the rotation matrix for the Gaussian particle as $\boldsymbol{R} = [\boldsymbol{t}_u, \boldsymbol{t}_v, \boldsymbol{t}_w]$. Moreover, along with photometric reconstruction, 2DGS additionally incorporates depth distortion and normal consistency to further enhance the quality of the reconstructions. For further details regarding 2DGS, please refer to [23].

In our approach, along with optimizing the parameters of each 2D Gaussian, we aim to jointly refine the keyframe poses as well. We do so by propagating the gradients of losses through the projection operation of the 2D Gaussians onto the image plane of each keyframe, as done in [26, 41, 81]. We use automatic differentiation via PyTorch [51] for calculating the gradients and updating the keyframe poses. Further details are in Appendix.

3.3. Dynamic Keyframe Selection for Gaussian Splatting Optimization

Once we obtain a coarse pose initialization of keyframes, we aim to construct a 2DGS model of the object. However, errors in the pose initialization can cause a divergence in the Gaussian Splatting optimization. Unlike BundleSDF’s ray-casting method [77], which renders individual pixels, Gaussian Splatting uses tile-based rasterization, rendering entire images one at a time, thereby increasing the computational cost linearly as the number of keyframes increases. To mitigate these issues, we introduce a dynamic keyframe selection approach to filter out erroneous keyframes.

To acquire a reliable Gaussian Object Field, we strategically sparsely select keyframes to optimize keyframe poses and object Gaussians. We first establish a series of “anchor points” at varying resolution levels, using the vertices and face centers of an icosahedron (as shown in Fig. 2-bottom left) to approximate evenly distributed points on a sphere centered on the object [58]. We then cluster the initial coarse keyframe pose estimates around these anchor points along the icosahedron. To maximize information from all viewpoints around the object, we select the keyframe with the largest object mask in the cluster of each anchor point, effectively training under sparse-view conditions with the aid of depth information [32]. This ensures that we minimize instances where the object is largely occluded and consider views where we have better visibility of the object.

While jointly optimizing the 2D Gaussians and the selected keyframe poses, we further remove outliers with erroneous pose estimates based on the reconstruction error obtained during the 2D Gaussian optimization. This approach is necessary because reconstruction residuals can impede pose optimization during the joint optimization of 2D Gaussians and keyframe poses. Specifically, we estimate the median absolute deviation (MAD) of the reconstruction loss at each iteration, which represents the typical “spread” around the median value, to identify and remove outlier views. The rationale for using MAD lies in its robustness; as a median-based metric, MAD is less influenced by extreme values than other measures, such as the mean or standard deviation, making it more reliable in the presence of outliers. Views with absolute deviations exceeding three times the MAD are classified as outliers.

3.4. Opacity Percentile-based Adaptive Density Control

During the optimization of the Gaussian Object Field, we perform periodic pruning and densification to maintain both the number and compactness of Gaussians. However, the vanilla Adaptive Density Control proposed in 3DGS has several limitations [2], since it demands considerable engineering work to adjust densification intervals and fine-tune opacity thresholds to stabilize training. Prior work [8] demonstrates that

a gradual iterative pruning strategy can yield significantly sparser models while preserving high fidelity. Similarly, Fan et al. [11] propose an importance weighting based on the scale percentile and the opacity of the Gaussians. However, they mainly focus on efficient compression of Gaussians. Inspired by [11], and since we have an object-centric focus, we limit the scale of the Gaussians and instead use a percentile-based pruning strategy based on opacity for stabilizing the number of Gaussians.

After a fixed number of optimization steps, we prune the Gaussians with opacity in the bottom 5th percentile until the opacity of the 95th percentile of the Gaussian particles exceeds a given threshold. This allows us to ensure that during the forward rendering (Eq. 1), a good number of high-quality Gaussian particles remain and those that are more inconsequential get pruned out. We empirically verify that our approach compared to naive absolute thresholding [27], improves the performance of our method. We further trigger splitting and cloning of the Gaussian particles when the positional gradient exceeds a predefined threshold, similar to [27]. Notably, the variation of the positional gradient remains relatively stable and does not continuously increase during training. Once the optimization of the Gaussian Object Field converges, the poses of all the keyframes are refined using the reconstruction of the RGB, depth, and normals, by temporarily freezing the 2D Gaussians.

3.5. Online Pose Graph Optimization

When we receive the updated poses for the keyframes from the Gaussian Object Field, we establish a global object-centric coordinate system and a keyframe memory pool, which stores key correspondences. To balance computational efficiency with memory usage and reduce long-term tracking drift when a new frame is observed, a set of overlapping frames from the memory pool is selected for graph optimization based on the view frustum of the incoming frame. For each frame in this pool, we generate a point-normal map and compute the dot-product of the normals with the camera-ray direction of the new frame, to assess visibility. Frames are selected if the visibility ratio of the new incoming frame exceeds a defined threshold. We choose the best keyframes from the pool to construct the pose graph along with the incoming frame. We optimize the pose graph using pairwise geometric consistency by minimizing a dense pixel-wise re-projection error as in [75].

4. Experiments

4.1. Datasets

4.1.1. YCBInEOAT Dataset

The YCBInEOAT dataset [79] offers ego-centric RGB-D video recordings of a dual-arm Yaskawa Motoman SDA10f robot manipulating YCB objects. Using an Azure Kinect

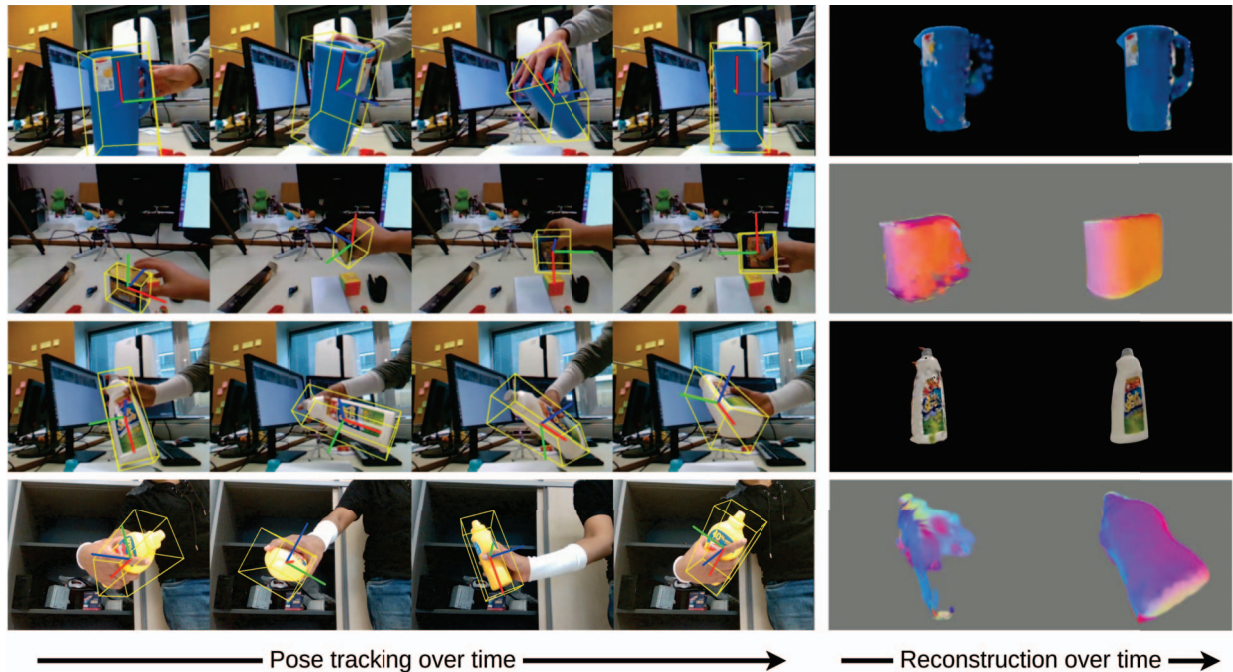


Figure 3. Qualitative results of our method, 6DOPE-GS, tested on video sequences from the HO3D dataset, namely AP13, MPM14, SB13, and SM1 (from top to bottom). **Left:** Our method tracks the 6D object pose over time with high accuracy, **Right:** 6DOPE-GS is effective at reconstructing both the appearance (rows 1 and 3) and surface geometry (rows 2 and 4) of the object over time. The first image shows the initial reconstruction at the beginning of the sequence, the second image shows the refined reconstruction over time.

camera positioned at mid-range, the dataset captures three types of manipulation tasks: single-arm pick-and-place, within-hand manipulation, and handoff between arms. In total, it includes 5 YCB objects [80] across 9 video sequences, amounting to 7,449 frames. Each frame is annotated with accurate 6D object poses, calibrated with the camera’s extrinsic parameters.

4.1.2. HO3D Dataset

The HO3D dataset [17] features 27 multi-view (68 single-view) sequences showing hand-object interactions involving 10 YCB objects [80] with annotated 3D poses. The RGB-D video data, captured at close range with an Intel RealSense camera, provides detailed records of hand manipulations of objects. Ground-truth 3D poses are generated via multi-view registration, facilitating evaluations of pose, reconstruction, and texture accuracy. We use the latest version, HO3D_v3, and conduct evaluations on the official test set, which comprises 4 objects and 13 sequences. Compared to YCBInEOAT, HO3D introduces increased complexity due to articulated hand-object interactions and rapid motion.

4.2. Metrics & Baselines

We evaluate the performance of different methods based on three key metrics: 6-DoF object pose tracking accuracy, 3D reconstruction accuracy, and computational effi-

ciency. Pose estimation accuracy is assessed using the area under the curve percentage for the ADD and ADD-S (ADD-Symmetric) metrics [21, 80]. For object reconstruction, we measure the Chamfer distance between the reconstructed and ground-truth object meshes in object coordinates. Computational efficiency is evaluated based on the average processing time per frame.

We compare several SLAM-based approaches, including DROID-SLAM (RGBD) [66], NICE-SLAM [84], KinectFusion [45], and SDF-2-SDF [60]. We report the performance of other approaches listed on the leaderboard of [77]. We evaluate recent Gaussian Splatting SLAM methods, including MonoGS [40] (3D Gaussians) and Endo2DTAM [24] (2D Gaussians), along with BundleTrack [75] and BundleSDF [77], using their open-source implementations^{2,3} with optimized parameters. For fair comparison, all methods utilize the same precomputed segmentation masks generated by XMem [6], consistent with the BundleSDF. MonoGS is evaluated under RGB-D input.

4.3. Results

As shown in Tables 1 and 2, our method outperforms SLAM-based baselines and BundleTrack [75]. In the YCBInEOAT dataset (Table 1), where object motions are relatively smooth

²<https://github.com/NVlabs/BundleSDF>

³<https://github.com/muskie82/MonoGS>

Method	Pose		Reconstruction	Efficiency
	ADD-S (%) \uparrow	ADD (%) \uparrow	CD (cm) \downarrow	ATPF(s) \downarrow
NICE-SLAM [84]	23.41	12.70	6.13	n.a.
SDF-2-SDF [60]	28.20	14.04	2.61	n.a.
DROID-SLAM [66]	46.39	34.68	4.63	n.a.
MaskFusion [57]	41.88	35.07	2.34	n.a.
MonoGS(RGB-D) [40]	20.16	15.32	2.43	0.29
Endo-2DTAM [24]	20.81	19.45	2.14	0.17
BundleTrack [75]	92.54	84.91	-	0.21
BundleSDF [77]	92.82	84.28	0.53	0.82
6DOPE-GS	93.79	87.82	0.15	0.22

Table 1. Comparison on the YCBInEoAT Dataset. ADD and ADD-S are reported as AUC percentages (0 to 0.1m), and reconstruction accuracy is measured by Chamfer loss. ATPF is the average processing time per frame (n.a. indicates unavailable data).

Method	Pose		Reconstruction	Efficiency
	ADD-S (%) \uparrow	ADD (%) \uparrow	CD (cm) \downarrow	ATPF(s) \downarrow
NICE-SLAM [84]	22.29	8.97	52.57	n.a.
SDF-2-SDF [60]	35.88	16.08	9.65	n.a.
KinectFusion [45]	25.81	16.54	15.49	n.a.
DROID-SLAM [66]	64.64	33.36	30.84	n.a.
MonoGS(RGB-D) [40]	2.81	1.82	22.09	0.36
Endo-2DTAM [24]	18.54	13.49	4.29	0.21
BundleTrack [75]	93.96	77.75	-	0.29
BundleSDF [77]	94.86	89.56	0.58	2.10
6DOPE-GS	95.07	84.33	0.41	0.24

Table 2. Comparison on the HO3D Dataset. ADD and ADD-S are reported as AUC percentages (0 to 0.1m), and reconstruction accuracy is measured by Chamfer loss. ATPF is the average processing time per frame (n.a. indicates unavailable data).

with less viewpoint diversity, most approaches perform similarly due to the absence of large occlusions or abrupt motion discontinuities that could lead to erroneous coarse pose initialization. In contrast, Gaussian Splatting-based methods such as MonoGS [40] and Endo-2DTAM [24] perform sub-optimally, due to low texture, occlusion, and the absence of pairwise keyframe optimization. On the HO3D dataset (Table 2), which presents more challenging scenarios with complex hand-object interactions and rapid motion variations, all baselines struggle with accurate tracking due to accumulating errors. In contrast, our pose graph optimization and keyframe selection enhance robustness in the coarse pose initialization and pose tracking efficiency, which further results in superior reconstruction at a sub-centimeter level. 6DOPE-GS maintains a strong balance between pose accuracy and temporal efficiency, making it well-suited for real-time applications.

While 6DOPE-GS outperforms BundleSDF [77] in ADD-S, it still lags in absolute accuracy on the more challenging HO3D dataset. A likely cause is severe occlusion in HO3D, which limits supervision for optimizing Gaussian Particles and iterative pose refinement. In contrast, BundleSDF benefits from mini-batch SDF rendering, enabling more effective correlated updates. However, its high computational cost hinders real-time use. Our method offers a more favorable trade-off between speed and accuracy, making it practical for real-world deployment. Qualitative results are shown in Fig. 3.

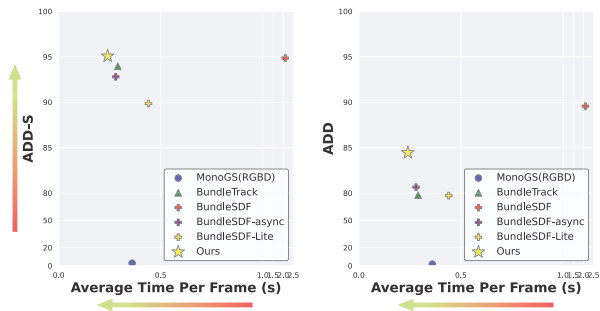


Figure 4. Comparison between temporal efficiency and performance for different approaches on the HO3D dataset. While BundleSDF achieves high performance, it comes at the cost of speed. On the other hand, 6DOPE-GS achieves a favorable tradeoff between speed and performance.

4.4. Temporal Efficiency

To evaluate the temporal efficiency of different approaches, we compare the tradeoff between the performance and the average processing time per frame for the different approaches on the HO3D dataset. We test the approaches on a desktop with a 12th Gen Intel(R) Core(TM) i9-12900KF CPU, 64GB RAM, equipped with an NVIDIA GeForce RTX 4090 GPU. We explore the performance of two more versions of BundleSDF [77] that reduce the processing time. In the first variant, “BundleSDF-async”, we disable synchronization between neural object field learning and online pose graph optimization (OPGO). Optimization of the neural field terminates once OPGO completes, improving runtime at the cost of reduced pose accuracy. In another variant, “BundleSDF-lite”, we reduce the number of optimization steps for learning the neural object field, enabling faster synchronization between the threads.

From Fig. 4, we observe that the high pose tracking accuracy of BundleSDF [77] comes at a high computational cost. Since the pose tracking thread waits for the neural object field thread to converge and subsequently synchronize the object pose and reconstruction estimates, it requires an average processing time of 2.1 seconds. Surprisingly, BundleSDF-async (without the sync between the threads) achieves better performance than BundleSDF-lite even though BundleSDF-async runs the pose estimation without waiting for the neural object field. This highlights the dependence of the pose graph optimization on accurate keyframe poses. While the neural object field training in BundleSDF-async yields more accurate poses (although at a delayed timestep) than BundleSDF-lite, the pose estimation of the latter diverges given the premature termination of the optimization to achieve faster speeds. In contrast, 6DOPE-GS provides a balanced trade-off between speed and accuracy. We achieve competitive performance without having to compromise on speed ($\sim 5\times$ speedup over BundleSDF) as a result of the rapid convergence of the Gaussian Object Field optimization.

4.5. Ablation Study

We assessed our design choices on both the HO3D and YCBInEoat dataset, chosen for its variety of scenarios, the results of which are shown in Table 3. *Ours (basic)* is a simplified version of 6DOPE-GS that naively uses all keyframes and employs vanilla adaptive density control. *Ours w/o KF Selection* removes the dynamic keyframe selection strategy (Sec. 3.3) and performs joint optimization using all keyframes. *Ours w/o Pruning* replaces the Opacity Percentile-based Adaptive Density Control 3.4 and employs vanilla adaptive density control [27]. We also compare 2DGS and 3DGS representation, *Ours (3DGS)* replaces the 2D Gaussian representation with a 3D Gaussian representation for pose estimation and reconstruction.

Performance was reduced without dynamic keyframe selection (*Ours w/o KF selection*) due to the retention of inaccurate pose estimates during training, which introduces residual errors in the reconstruction loss and hinders pose optimization. Applying the vanilla adaptive density control (*Ours w/o Pruning*), where all Gaussians below a predefined threshold are removed, causes abrupt changes in the number of Gaussians. This results in significant rendering fluctuations, slowing the convergence of training. The pose accuracy and reconstruction quality of 3DGS (*Ours (3DGS)*) are inferior to 2DGS. This can be attributed to the lack of regularization on the normal and depth in 3DGS, causing the Gaussians to deviate from the object surface and consequently degrading the reconstruction quality. We find that our approach with the proposed additions, namely Dynamic Keyframe Selection and the Opacity Percentile-based Adaptive Density Control performs the best among all.

	Method	Pose		Reconstruction
		ADD-S (%) \uparrow	ADD (%) \uparrow	CD (cm) \downarrow
HO3D	Ours (basic)	93.52	80.25	0.44
	Ours w/o KF Selection	94.44	82.40	0.42
	Ours w/o Pruning	92.48	80.87	0.44
	Ours (3DGS)	92.51	79.49	0.47
	Ours (final)	95.07	84.33	0.41
YCBInEoat	Ours (basic)	92.74	85.15	0.22
	Ours w/o KF Selection	93.03	86.40	0.19
	Ours w/o Pruning	92.64	86.22	0.20
	Ours (3DGS)	91.18	85.29	0.41
	Ours (final)	93.79	87.82	0.15

Table 3. Ablation study of critical design choices

4.6. Realtime Results

We utilized the ZED 2 camera operating in the standard depth sensing mode to maintain a balance between frame rate and accuracy. The camera captures video at a resolution of 1080p with a frame rate of 30 FPS. An initial mask for the target object was manually created through human annotation. The SAM2 system also operates at 28 FPS. Pose tracking, when running in visualization mode, achieves a processing frequency of 3-4 Hz, primarily due to the compu-

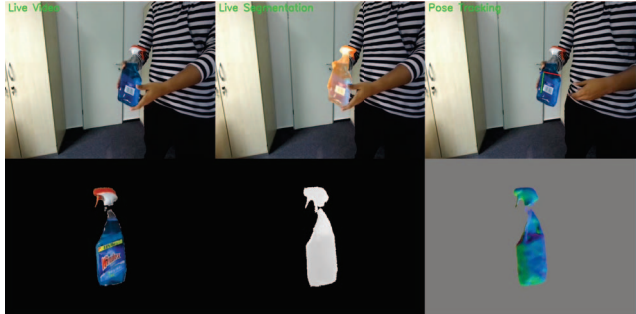


Figure 5. Example of real-time object tracking. **Top row:** Live video, object segmentation results, and pose tracking results. **Bottom row:** Rendered outputs, including color, depth, and surface normals derived from the Gaussian models.

tational overhead introduced by the GUI and the rendering of Gaussian models in the background. Without the GUI, the system can operate at a slightly higher frequency of 4-5 Hz. The Gaussian model updates approximately every 8 seconds, as illustrated in Fig 5. For a more comprehensive understanding of the system’s performance, we encourage readers to refer to the supplementary video provided.

5. Conclusion

In this paper, we proposed “6DOPE-GS”, a novel method for model-free 6D object pose estimation and reconstruction that leveraged 2D Gaussian Splatting for jointly optimizing object pose estimates and 3D reconstruction in an iterative manner. Key to our method’s efficiency were a novel dynamic keyframe selection mechanism based on spatial coverage, as well as a confidence-based filtering mechanism to remove erroneous keyframes, followed by an opacity percentile-based adaptive density control for pruning out inconsequential Gaussians. These contributions enabled 6DOPE-GS to achieve competitive performance in a computationally efficient manner ($\sim 5\times$ speedup), as validated on the HO3D and YCBInEoat datasets, successfully capturing a practical balance of speed, accuracy, and stability for dynamic tracking scenarios in near real-time.

However, some shortcomings remain, which we aim to address in future work. Although Gaussian rasterization rendering is highly efficient and allows for rapid refinement of small translation and in-plane rotation errors, it may be less effective in gradient computations compared to differentiable ray casting used by neural radiance fields. To address this, we plan to investigate ray casting for rendering Gaussian representations [16, 44], which could improve both performance and computational efficiency. Another potential limitation is that the optimized 2D Gaussians are not directly integrated into the online pose graph optimization; instead, only the optimized poses are used. In future work, we will explore ways to more closely couple the trained object representation with the pose graph optimization.

References

- [1] K Somani Arun, Thomas S Huang, and Steven D Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on pattern analysis and machine intelligence*, (5):698–700, 1987. 4
- [2] Samuel Rota Bulò, Lorenzo Porzi, and Peter Kontschieder. Revising Densification in Gaussian Splatting. 5
- [3] Jianqiu Chen, Zikun Zhou, Mingshan Sun, Rui Zhao, Liwei Wu, Tianpeng Bao, and Zhenyu He. Zeropose: Cad-prompted zero-shot object 6d pose estimation in cluttered scenes. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024. 2
- [4] Kai Chen and Qi Dou. Sgpa: Structure-guided prior adaptation for category-level 6d object pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2773–2782, 2021. 2
- [5] Weifeng Chen, Guangtao Shang, Aihong Ji, Chengjun Zhou, Xiyang Wang, Chonghui Xu, Zhenxiong Li, and Kai Hu. An overview on visual slam: From tradition to semantic. *Remote Sensing*, 14(13):3010, 2022. 3
- [6] Ho Kei Cheng and Alexander G. Schwing. XMem: Long-Term Video Object Segmentation with an Atkinson-Shiffrin Memory Model. 6
- [7] Julian Chibane, Gerard Pons-Moll, et al. Neural unsigned distance fields for implicit function learning. *Advances in Neural Information Processing Systems*, 33:21638–21652, 2020. 3
- [8] Chenxi Lola Deng and Enzo Tartaglione. Compressing explicit voxel grid representations: fast nerfs become also small. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1236–1245, 2023. 5
- [9] Xinke Deng, Arsalan Mousavian, Yu Xiang, Fei Xia, Timothy Bretl, and Dieter Fox. PoseRBPF: A Rao–Blackwellized Particle Filter for 6-D Object Pose Tracking. 37(5):1328–1342. 2
- [10] Xinke Deng, Yu Xiang, Arsalan Mousavian, Clemens Eppner, Timothy Bretl, and Dieter Fox. Self-supervised 6d object pose estimation for robot manipulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3665–3671. IEEE, 2020. 1
- [11] Zhiwen Fan, Kevin Wang, Kairun Wen, Zehao Zhu, Dejie Xu, and Zhangyang Wang. Lightgaussian: Unbounded 3d gaussian compression with 15x reduction and 200+ fps. *arXiv preprint arXiv:2311.17245*, 2023. 5
- [12] Martin A Fischler and Robert 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. 4
- [13] Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9(1-2):1–148, 2015. 2
- [14] Ge Gao, Mikko Lauri, Yulong Wang, Xiaolin Hu, Jianwei Zhang, and Simone Frntrop. 6d object pose regression via supervised learning on point clouds. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3643–3649. IEEE, 2020. 2
- [15] Felix Gorschlüter, Pavel Rojtbeg, and Thomas Pöllabauer. A survey of 6d object detection based on 3d models for industrial applications. *Journal of Imaging*, 8(3):53, 2022. 1
- [16] Chun Gu, Xiaofei Wei, Zixuan Zeng, Yuxuan Yao, and Li Zhang. IRGS: Inter-Reflective Gaussian Splatting with 2D Gaussian Ray Tracing. 8
- [17] Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. Honnotate: A method for 3d annotation of hand and object poses. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3196–3206, 2020. 6
- [18] Xingyi He, Jiaming Sun, Yuang Wang, Di Huang, and Xiaowei Zhou. OnePose++: Keypoint-Free One-Shot Object Pose Estimation without CAD Models. . 2
- [19] Yisheng He, Haibin Huang, Haoqiang Fan, Qifeng Chen, and Jian Sun. FFB6D: A Full Flow Bidirectional Fusion Network for 6D Pose Estimation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3002–3012. IEEE, . 2
- [20] Yisheng He, Wei Sun, Haibin Huang, Jianran Liu, Haoqiang Fan, and Jian Sun. PVN3D: A Deep Point-Wise 3D Keypoints Voting Network for 6DoF Pose Estimation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11629–11638. IEEE, . 2
- [21] Yisheng He, Yao Wang, Haoqiang Fan, Jian Sun, and Qifeng Chen. FS6D: Few-Shot 6D Pose Estimation of Novel Objects. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6804–6814. IEEE, . 2, 6
- [22] Yinlin Hu, Pascal Fua, Wei Wang, and Mathieu Salzmann. Single-stage 6d object pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2930–2939, 2020. 2
- [23] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2D Gaussian Splatting for Geometrically Accurate Radiance Fields. 2, 3, 4
- [24] Yiming Huang et al. Advancing dense endoscopic reconstruction with gaussian splatting-driven surface normal-aware tracking and mapping. *arXiv preprint arXiv:2501.19319*, 2025. 6, 7
- [25] Iman Abaspor Kazerouni, Luke Fitzgerald, Gerard Dooly, and Daniel Toal. A survey of state-of-the-art on visual slam. *Expert Systems with Applications*, 205:117734, 2022. 3
- [26] Nikhil Keetha, Jay Karhade, Krishna Murthy Jatavallabhula, Gengshan Yang, Sebastian Scherer, Deva Ramanan, and Jonathon Luiten. Splatam: Splat track & map 3d gaussians for dense rgb-d slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21357–21366, 2024. 2, 3, 4
- [27] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. 2, 3, 4, 5, 8
- [28] Kilian Kleiberger, Christian Landgraf, and Marco F Huber. Large-scale 6d object pose estimation dataset for industrial bin-picking. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2573–2578. IEEE, 2019. 1

- [29] Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. Cosypose: Consistent multi-view multi-object 6d pose estimation. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVII 16*, pages 574–591. Springer, 2020. 2
- [30] Yann Labbé, Lucas Manuelli, Arsalan Mousavian, Stephen Tyree, Stan Birchfield, Jonathan Tremblay, Justin Carpentier, Mathieu Aubry, Dieter Fox, and Josef Sivic. Megapose: 6d pose estimation of novel objects via render & compare. *arXiv preprint arXiv:2212.06870*, 2022. 2
- [31] Hongyu Li, Snehal Dikhale, Soshi Iba, and Nawid Jamali. Vihope: Visuotactile in-hand object 6d pose estimation with shape completion. *IEEE Robotics and Automation Letters*, 8(11):6963–6970, 2023. 2
- [32] Yanyan Li, Chenyu Lyu, Yan Di, Guangyao Zhai, Gim Hee Lee, and Federico Tombari. Geogaussian: Geometry-aware gaussian splatting for scene rendering. In *European Conference on Computer Vision*, pages 441–457. Springer, 2025. 5
- [33] Zechu Li, Yufeng Jin, Daniel Ordonez Apraetz, Claudio Semini, Puze Liu, and Georgia Chalvatzaki. Morphologically symmetric reinforcement learning for ambidextrous bimanual manipulation. *arXiv preprint arXiv:2505.05287*, 2025. 1
- [34] Jiehong Lin, Lihua Liu, Dekun Lu, and Kui Jia. SAM-6D: Segment Anything Model Meets Zero-Shot 6D Object Pose Estimation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 27906–27916. IEEE, 2024. 2
- [35] Xingyu Liu, Shun Iwase, and Kris M Kitani. Kdfnet: Learning keypoint distance field for 6d object pose estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4631–4638. IEEE, 2021. 2
- [36] Yuan Liu, Yilin Wen, Sida Peng, Cheng Lin, Xiaoxiao Long, Taku Komura, and Wenping Wang. Gen6D: Generalizable Model-Free 6-DoF Object Pose Estimation from RGB Images. 2
- [37] Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *2024 International Conference on 3D Vision (3DV)*, pages 800–809. IEEE, 2024. 3
- [38] Andréa Macario Barros, Maugan Michel, Yoann Moline, Gwenolé Corre, and Frédérick Carrel. A comprehensive survey of visual slam algorithms. *Robotics*, 11(1):24, 2022. 3
- [39] Simon Manschitz, Berk Gueler, Wei Ma, and Dirk Ruiken. Sampling-based grasp and collision prediction for assisted teleoperation. *arXiv preprint arXiv:2504.18186*, 2025. 1
- [40] Hidenobu Matsuki, Riku Murai, Paul H. J. Kelly, and Andrew J. Davison. Gaussian Splatting SLAM. 6, 7
- [41] Hidenobu Matsuki, Riku Murai, Paul HJ Kelly, and Andrew J Davison. Gaussian splatting slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18039–18048, 2024. 3, 4
- [42] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 3
- [43] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 3
- [44] Nicolas Moenne-Loccoz, Ashkan Mirzaei, Or Perel, Riccardo de Lutio, Janick Martinez Esturo, Gavriel State, Sanja Fidler, Nicholas Sharp, and Zan Gojcic. 3d gaussian ray tracing: Fast tracing of particle scenes. *arXiv preprint arXiv:2407.07090*, 2024. 8
- [45] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE international symposium on mixed and augmented reality*, pages 127–136. Ieee, 2011. 6, 7
- [46] Van Nguyen Nguyen, Thibault Groueix, Mathieu Salzmann, and Vincent Lepetit. GigaPose: Fast and Robust Novel Object Pose Estimation via One Correspondence. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9903–9913. IEEE, 2024. 2
- [47] Joseph Ortiz, Alexander Clegg, Jing Dong, Edgar Sucar, David Novotny, Michael Zollhoefer, and Mustafa Mukadam. isdf: Real-time neural signed distance fields for robot perception. In *Robotics: Science and Systems*, 2022. 3
- [48] Onur Özyeşil, Vladislav Voroninski, Ronen Basri, and Amit Singer. A survey of structure from motion*. *Acta Numerica*, 26:305–364, 2017. 2
- [49] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 3
- [50] Keunhong Park, Arsalan Mousavian, Yu Xiang, and Dieter Fox. Latentfusion: End-to-end differentiable reconstruction and rendering for unseen object pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10710–10719, 2020. 2
- [51] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 4
- [52] Georgios Pavlakos, Xiaowei Zhou, Aaron Chan, Konstantinos G. Derpanis, and Kostas Daniilidis. 6-DoF object pose from semantic keypoints. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2011–2018. 2
- [53] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. PVNet: Pixel-Wise Voting Network for 6DoF Pose Estimation. pages 4561–4570. 2
- [54] Mahdi Rad and Vincent Lepetit. Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *Proceedings of the IEEE international conference on computer vision*, pages 3828–3836, 2017. 2
- [55] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman

- Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment Anything in Images and Videos. [3](#), [4](#)
- [56] Xinlin Ren, Xingkui Wei, Zhuwen Li, Yanwei Fu, Yinda Zhang, and Xiangyang Xue. Deepsfm: Robust deep iterative refinement for structure from motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(6):4058–4074, 2023. [3](#)
- [57] Martin Runz, Maud Buffier, and Lourdes Agapito. Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects. In *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 10–20. IEEE, 2018. [7](#)
- [58] Edward B Saff and Amo BJ Kuijlaars. Distributing many points on a sphere. *The mathematical intelligencer*, 19:5–11, 1997. [5](#)
- [59] Ivan Shugurov, Fu Li, Benjamin Busam, and Slobodan Ilic. Osop: A multi-stage one shot object pose estimation framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6835–6844, 2022. [2](#)
- [60] Miroslava Slavcheva, Wadim Kehl, Nassir Navab, and Slobodan Ilic. Sdf-2-sdf: Highly accurate 3d object reconstruction. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 680–696. Springer, 2016. [6](#), [7](#)
- [61] Stefan Stevšić, Sammy Christen, and Otmar Hilliges. Learning to assemble: Estimating 6d poses for robotic object-object manipulation. *IEEE Robotics and Automation Letters*, 5(2): 1159–1166, 2020. [1](#)
- [62] Yongzhi Su, Jason Rambach, Nareg Minaskan, Paul Lesur, Alain Pagani, and Didier Stricker. Deep multi-state object pose estimation for augmented reality assembly. In *2019 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pages 222–227. IEEE, 2019. [1](#)
- [63] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. LoFTR: Detector-Free Local Feature Matching with Transformers. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8918–8927. IEEE. [3](#), [4](#)
- [64] Jiaming Sun, Zihao Wang, Siyu Zhang, Xingyi He, Hongcheng Zhao, Guofeng Zhang, and Xiaowei Zhou. Onepose: One-shot object pose estimation without cad models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6825–6834, 2022. [2](#)
- [65] Martin Sundermeyer, Zoltan-Csaba Marton, Maximilian Durner, Manuel Brucker, and Rudolph Triebel. Implicit 3d orientation learning for 6d object detection from rgb images. In *Proceedings of the european conference on computer vision (ECCV)*, pages 699–715, 2018. [2](#)
- [66] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *Advances in neural information processing systems*, 34:16558–16569, 2021. [6](#), [7](#)
- [67] Bugra Tekin, Sudipta N. Sinha, and Pascal Fua. Real-Time Seamless Single Shot 6D Object Pose Prediction. pages 292–301. [2](#)
- [68] Chen Wang, Roberto Martín-Martín, Danfei Xu, Jun Lv, Cewu Lu, Li Fei-Fei, Silvio Savarese, and Yuke Zhu. 6-PACK: Category-level 6D Pose Tracker with Anchor-Based Keypoints. [2](#)
- [69] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martin-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese. DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion. pages 3343–3352. [2](#)
- [70] Chao Wang, Anna Belardinelli, Stephan Hasler, Theodoros Stouraitis, Daniel Tanneberg, and Michael Gienger. Explainable human-robot training and cooperation with augmented reality. In *Extended abstracts of the 2023 CHI conference on human factors in computing systems*, pages 1–5, 2023. [1](#)
- [71] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. *arXiv preprint arXiv:2408.16061*, 2024. [3](#)
- [72] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J. Guibas. Normalized Object Coordinate Space for Category-Level 6D Object Pose and Size Estimation. [2](#)
- [73] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024. [3](#)
- [74] Xiang Wang, Chen Wang, Bing Liu, Xiaoqing Zhou, Liang Zhang, Jin Zheng, and Xiao Bai. Multi-view stereo in the deep learning era: A comprehensive review. *Displays*, 70: 102102, 2021. [2](#)
- [75] Bowen Wen and Kostas Bekris. BundleTrack: 6D Pose Tracking for Novel Objects without Instance or Category-Level 3D Models. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8067–8074. [2](#), [3](#), [4](#), [5](#), [6](#), [7](#)
- [76] Bowen Wen, Chaitanya Mitash, Baozhang Ren, and Kostas E. Bekris. Se(3)-TrackNet: Data-driven 6D Pose Tracking by Calibrating Image Residuals in Synthetic Domains. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10367–10373. [2](#)
- [77] Bowen Wen, Jonathan Tremblay, Valts Blukis, Stephen Tyree, Thomas Muller, Alex Evans, Dieter Fox, Jan Kautz, and Stan Birchfield. BundleSDF: Neural 6-DoF Tracking and 3D Reconstruction of Unknown Objects. [2](#), [3](#), [4](#), [5](#), [6](#), [7](#)
- [78] Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. FoundationPose: Unified 6D Pose Estimation and Tracking of Novel Objects. [2](#)
- [79] Bowen Wen, Chaitanya Mitash, Baozhang Ren, and Kostas E Bekris. se (3)-tracknet: Data-driven 6d pose tracking by calibrating image residuals in synthetic domains. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10367–10373. IEEE, 2020. [5](#)
- [80] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017. [6](#)
- [81] Chi Yan, Delin Qu, Dan Xu, Bin Zhao, Zhigang Wang, Dong Wang, and Xuelong Li. Gs-slam: Dense visual slam with 3d gaussian splatting. In *Proceedings of the IEEE/CVF Con-*

- ference on Computer Vision and Pattern Recognition*, pages 19595–19604, 2024. [3](#), [4](#)
- [82] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. *arXiv preprint arXiv:2410.03825*, 2024. [3](#)
- [83] Yan Zhao, Shaobo Zhang, Wanqing Zhao, Ying Wei, and Jinye Peng. Augmented reality system based on real-time object 6d pose estimation. In *2023 2nd International Conference on Image Processing and Media Computing (ICIPMC)*, pages 27–34. IEEE, 2023. [1](#)
- [84] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12786–12796, 2022. [6](#), [7](#)
- [85] Evin Pınar Örnek, Yann Labbé, Bugra Tekin, Lingni Ma, Cem Keskin, Christian Forster, and Tomas Hodan. FoundPose: Unseen Object Pose Estimation with Foundation Features. [2](#)