

Object Gaussian for Monocular 6D Pose Estimation from Sparse Views

Luqing Luo^{*1}, Shichu Sun^{*1}, Jiangang Yang¹, Linfang Zheng², Jinwei Du³, Jian Liu^{†1}

¹Institute of Microelectronics Chinese Academy of Sciences, Beijing, China

²University of Birmingham, Birmingham, UK

³NVIDIA, Shanghai, China

Abstract

Monocular object pose estimation, as a pivotal task in computer vision and robotics, heavily depends on accurate 2D-3D correspondences, which often demand costly CAD models that may not be readily available. Object 3D reconstruction methods offer an alternative, among which recent advancements in 3D Gaussian Splatting (3DGS) afford a compelling potential. Yet its performance still suffers and tends to overfit with fewer input views. Embracing this challenge, we introduce SGPose, a novel framework for sparse view object pose estimation using Gaussian-based methods. Given as few as ten views, SGPose generates a geometric-aware representation by starting with a random cuboid initialization, eschewing reliance on Structure-from-Motion (SfM) pipeline-derived geometry as required by traditional 3DGS methods. SGPose removes the dependence on CAD models by regressing dense 2D-3D correspondences between images and the reconstructed model from sparse input and random initialization, while the geometric-consistent depth supervision and online synthetic view warping are key to the success. Experiments on typical benchmarks, especially on the Occlusion LM-O dataset, demonstrate that SGPose outperforms existing methods even under sparse view constraints, under-scoring its potential in real-world applications.

Introduction

Monocular pose estimation in 3D space, while inherently ill-posed, is a necessary step for many tasks involving human-object interactions, such as robotic grasping and planning (Azad, Asfour, and Dillmann 2007), augmented reality (Tan, Tombari, and Navab 2018), and autonomous driving (Manhardt, Kehl, and Gaidon 2019; Qi et al. 2018). Influenced by deep learning approaches, its evolution has enabled impressive performance even in cluttered environments. The most studied task in this field assumes that the CAD model of the object is known a priori (Peng et al. 2019; Li, Wang, and Ji 2019; Park, Patten, and Vincze 2019; Cai and Reid 2020; Chen et al. 2020; Park et al. 2020), but the accessibility of such a predefined geometry information prevents its applicability in real-world settings. To reduce reliance on specific object CAD models, recent research has shifted toward category-level pose estimation (Wang et al. 2019; Ahmadyan et al. 2021), aiming to generalize across

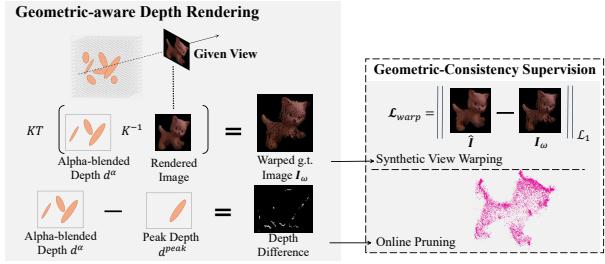


Figure 1: The alpha-blended depth d^α integrates depth across Gaussian primitives along the ray, the peak depth d^{peak} selects the one of highest opacity. d^α enables reliable online synthetic view warping without leveraging external depth information, in conjunction with d^{peak} guides the online pruning, both of them contribute to object Gaussian reconstruction under sparse views.

objects within the same category. However, these methods typically ask for extra depth information, and can falter with instances of varying appearances.

Emerging real-world demands call for an object pose estimator generalizable, flexible, and computationally efficient. Ideally, a new object can be reconstructed from casually taken reference images, sans the need for fine-grained well-textured 3D structures. Reconstruction-based methods have shown the feasibility of this proposal (Cai and Reid 2020; Liu et al. 2022b; Sun et al. 2022; He et al. 2022; Li et al. 2023; Cai, Heikkilä, and Rahtu 2024), which basically reconstruct the 3D object from the multi-view RGB images to substitute missing CAD model. however, reconstruction-based methods have long relied on a fixed budget of high-quality given images and the prerequisite use of Structure-from-Motion (SfM) techniques, resulting in a notably tedious and costly training process. Our method deviates from these requirements by pioneering an efficient object reconstruction method that thrives on limited reference images and the convenience of random initialization. Capitalizing on the high-quality scene representation and real-time rendering of 3DGS (Kerbl et al. 2023), we unveil SGPose, an novel framework of Sparse View Object Gaussian for monocular 6D Pose estimation. The proposed SGPose develops geometric-aware depth to guide object-centric 3D

^{*}These authors contributed equally.
Copyright © 2025, All rights reserved.

representations from RGB only input. Requiring a mere ten views, SGPose achieves competitive performance for object pose estimation, heralding its readiness for real-world deployment.

In our work, we extend a variant of 3DGS (Huang et al. 2024) by formulating Gaussian primitives as elliptic disks instead of ellipsoids to derive depth rendering. As illustrated in Fig. 1, the conceived geometric-consistent constraints guide depth acquisition. The resulting depth enables reliable online synthetic view warping and eases the challenges of sparse views for traditional 3DGS methods. Additionally, an online pruning is incorporated in terms of the geometric-consistent depth supervision, toning down common sparse view reconstruction issues like floaters and background collapses. In contrast to the conventional reliance on SfM pipelines for point cloud initialization in 3DGS, the proposed SGPose opts for a random initialization from a cuboid of 4,096 points. The proposed object Gaussian generates dense image pixels and object coordinates correspondence (2D-3D correspondence) maps efficiently using geometric-aware depth rendering, serving as a keystone advantage for monocular 6D pose estimation. An adapted GDRNet++ framework (Liu et al. 2022a) is utilized to assess 6D pose estimation on the LM (Hinterstoisser et al. 2012) and Occlusion LM-O (Brachmann et al. 2014) datasets. Our SGPose takes sparse view images and pose annotations to create synthetic views, object masks, and dense correspondence maps. Noteworthy, for the Occlusion LM-O dataset, we render data similar to PBR (Physically Based Rendering) data (Denninger et al. 2023), which further enhance the performance of the proposed method. By matching state-of-the-art performance across CAD-based and CAD-free approaches, we highlight the efficiency and flexibility of our method. To sum up, Our main contributions are:

- By intaking only RGB images, the proposed geometric-aware object Gaussian derives accurate depth rendering from random point initialization;
- The rendered depth ensures a reliable synthetic view warping and an effective online pruning, addressing the issue of overfitting under sparse views at an impressively low time cost;
- By generating dense 2D-3D correspondences and images that simulate real occlusions using the proposed object Gaussian, our SGPose framework achieves CAD-free monocular pose estimation that is both efficient and robust.

Related Work

CAD-Based Object Pose Estimation Many previous works on pose estimation rely on known CAD models. Regression-based methods (Kehl et al. 2017; Labb   et al. 2020; Li et al. 2018; Xiang et al. 2017) estimate pose parameters directly from features in regions of interest (RoIs), while keypoint-based methods establish correspondences between 2D image pixels and 3D object coordinates either by regression (Oberweger, Rad, and Lepetit 2018; Park, Patten, and Vincze 2019; Pavlakos et al. 2017) or by voting (Peng et al. 2019), often solve poses by using a variant

of Perspective-n-Points (PnP) algorithms (Lepetit, Moreno-Noguer, and Fua 2009). NOCS (Wang et al. 2019) establishes correspondences between image pixels and Normalized Object Coordinates (NOCS) shared across a category, reducing dependency on CAD models at test time. Later works (Lee et al. 2021; Tian, Ang, and Lee 2020; Wang et al. 2021; Wang, Chen, and Dou 2021) build upon this idea by leveraging category-level priors to recover more accurate shapes. A limitation of these methods is that objects within the same category can have significant variations in shape and appearance, which challenges the generalization of trained networks. Additionally, accurate CAD models are required for generating ground-truth NOCS maps during training. In contrast, our framework reconstructs 3D object models from pose-annotated images, enabling CAD-free object pose estimation during both training and testing phases.

CAD-Free Object Pose Estimation Some endeavors have been made to relax the constraints of CAD models of the objects. RLLG (Cai and Reid 2020) uses multi-view consistency to supervise coordinate prediction by minimizing reprojection error. NeRF-Pose (Li et al. 2023) trains a NeRF-based (Mildenhall et al. 2021) implicit neural representation of object and regresses object coordinate for pose estimation. Gen6D (Liu et al. 2022b) initializes poses using detection and retrieval but requires accurate 2D bounding boxes and struggles with occlusions. GS-Pose (Cai, Heikkil  , and Rahtu 2024) improves on Gen6D (Liu et al. 2022b) by employing a joint segmentation method and 3DGS-based refinement. OnePose (Sun et al. 2022) reconstructs sparse point clouds of objects and extracts 2D-3D correspondences, though its performance is limited on symmetric or textureless objects due to its reliance on repeatable keypoint detection. While OnePose++ (He et al. 2022) removes the dependency on keypoints resulting in a performance enhancement. Unlike these methods, which require numerous input images for training, we directly leverage the power of 3DGS (Kerbl et al. 2023) for geometric-aware object reconstruction from sparse, pose-annotated images to achieve pose estimation.

Methods

An overview of the proposed method is presented in Fig. 2. Given sparse views as input, the dense 2D-3D correspondence maps are encoded in the conceived object Gaussian naturally, by supervising geometric-aware depth. Consequently, the created synthetic views and correspondence maps are availed to a downstream pose estimator, to achieve CAD-free monocular Pose Estimation.

Depth Rendering of Geometric-aware Object Gaussian

The object geometry is described by Gaussian primitives of probability density function as (Kerbl et al. 2023),

$$\mathcal{G}(\mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})}, \quad (1)$$

where \mathbf{x} is a point in world space to describe the target object and $\boldsymbol{\mu}$ is the mean of each Gaussian primitive (which also is the geometric center). Thus, the difference vector $\mathbf{x} - \boldsymbol{\mu}$

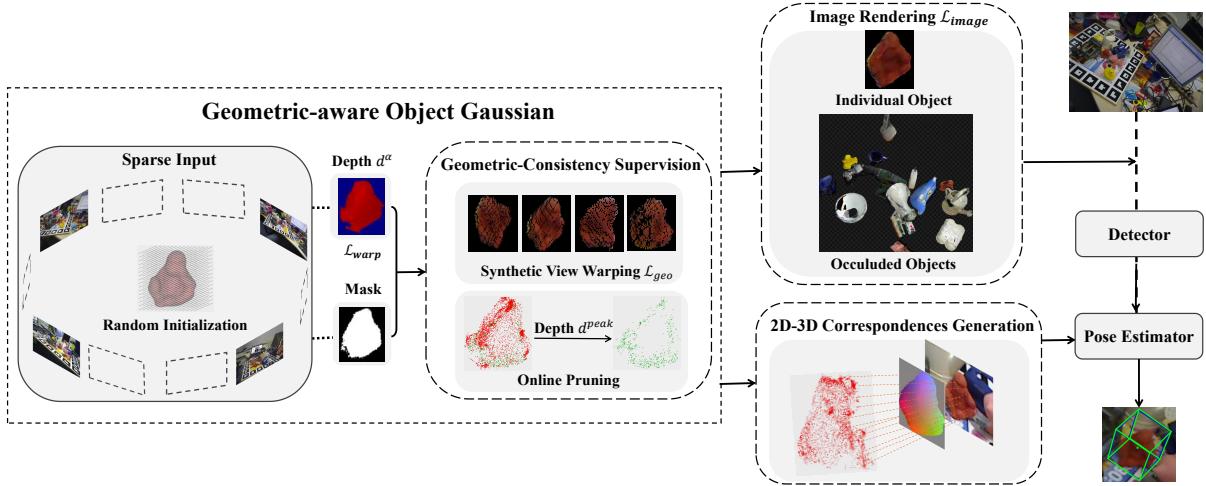


Figure 2: SGPose Pipeline. Given sparse RGB images and a **cuboid random initialization**, the object Gaussian learns the geometry of target objects under **the supervision of geometric-consistency**, to render synthetic views, including both of individual object images and occluded objects images, masks and dense 2D-3D correspondences. The image rendering loss \mathcal{L}_{image} , image warping loss \mathcal{L}_{warp} and geometric-consistent loss \mathcal{L}_{geo} are used to guide the learning process. For pose estimation, the objects are **detected and cropped from test images by detector** (Redmon and Farhadi 2018), the above rendering results, as the replacement of CAD models, are feed to pose estimator (Liu et al. 2022a) for regression.

indicts the probability density of \mathbf{x} , which peaks at the center μ and decreases as departing from it.

By treating Gaussian primitives as the **elliptical disks** (Huang et al. 2024), the **covariance matrix Σ** is parameterized on a local tangent plane centered at μ with a **rotation matrix** and a **scaling matrix**. Concretely, the rotation matrix R is comprised of three vectors $\mathbf{t}_u, \mathbf{t}_v$ and \mathbf{t}_w , where two orthogonal tangential vectors \mathbf{t}_u and \mathbf{t}_v indicate the orientations within the local tangent plane, and $\mathbf{t}_w = \mathbf{t}_u \times \mathbf{t}_v$ represents the **normal perpendicular to the plane**. The scaling matrix S depicts the variances of Gaussian primitives on corresponding directions, noted that there is no distribution in the direction of \mathbf{t}_w since the Gaussian primitive is defined on a flat elliptical disk. Thereby, the 3×3 rotation matrix $R = [\mathbf{t}_u, \mathbf{t}_v, \mathbf{t}_w]$ and the scaling matrix $S = [s_u, s_v, 0]$ form up the covariance matrix as $\Sigma = RSS^\top R^\top$.

By leveraging the world-to-camera transformation matrix W and the **Jacobian of the affine approximation of the projective transformation matrix J** , the projected 2D covariance matrix Σ' in camera coordinates is given as, $\Sigma' = JW\Sigma W^\top J^\top$. By virtue of the same structure and properties are maintained by **skipping the third row and column of Σ'** (Kopanas et al. 2021; Zwicker et al. 2001), a 2×2 variance matrix Σ^{2D} (corresponding to \mathcal{G}^{2D}) is obtained,

$$\mathcal{G}^{2D}(\mathbf{x}') = e^{-\frac{1}{2}(\mathbf{x}' - \mu')^\top (\Sigma^{2D})^{-1} (\mathbf{x}' - \mu')}, \quad (2)$$

where \mathbf{x}' and μ' stands for the projected points of \mathbf{x} and μ in the screen space, respectively.

Furthermore, the local tangent plane is defined as,

$$X(u, v) = \mu + s_u \mathbf{t}_u u + s_v \mathbf{t}_v v = \mathbf{H}(u, v, 1, 1)^\top, \quad (3)$$

where $\mathbf{H} = \begin{bmatrix} s_u \mathbf{t}_u & s_v \mathbf{t}_v & \mathbf{0} & \mu \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} RS & \mu \\ \mathbf{0} & 1 \end{bmatrix}$ is a homogeneous transformation matrix mapping point (u, v) on local tangent plane into the world space.

Suppose there is a ray emitting from the camera optical center onto the screen space. The geometric-aware depth d^{geo} is hence defined as the distance between the camera and the Gaussian primitive along the ray. Accordingly, the homogeneous coordinate of the point (u, v) projected onto the screen is (Zwicker et al. 2004),

$$(u', v', d, w)^\top = WX(u, v) = W\mathbf{H}(u, v, 1, 1)^\top. \quad (4)$$

where w is usually set to 1 (the homogeneous representation describes a point when $w \neq 0$, while it depicts a ray when $w = 0$). **This point can be further represented as the intersection of two orthogonal planes corresponding to u' and v'** (Weyrich et al. 2007; Sigg et al. 2006). Specifically, u' -plane is defined by a normal vector $(-1, 0, 0)$ and an offset u' , the 4D homogeneous plane thus is $\mathbf{h}_{u'} = (-1, 0, 0, u')$. Similarly, v' -plane is $\mathbf{h}_{v'} = (0, -1, 0, v')$. Conversely, both planes can be transformed back to the local tangent plane coordinates as,

$$\mathbf{h}_u = (W\mathbf{H})^\top \mathbf{h}_{u'}, \quad \mathbf{h}_v = (W\mathbf{H})^\top \mathbf{h}_{v'}, \quad (5)$$

in which $(W\mathbf{H})^\top$ is equivalent to $(W\mathbf{H})^{-1}$ as show in (Vince 2008). According to (Huang et al. 2024), since the screen point (u', v') must lie on both u' -plane and v' -plane, for any point $(u, v, 1, 1)$ on the elliptical disk, the dot product of the transformed plane \mathbf{h}_u and \mathbf{h}_v with the point $(u, v, 1, 1)$ should be zero,

$$\mathbf{h}_u \cdot (u, v, 1, 1)^\top = \mathbf{h}_v \cdot (u, v, 1, 1)^\top = 0, \quad (6)$$

by solving the equation above, the coordinates of the screen point (u', v') on the local tangent plane are yielded,

$$u = \frac{\mathbf{h}_u^2 \mathbf{h}_v^4 - \mathbf{h}_u^4 \mathbf{h}_v^2}{\mathbf{h}_u^1 \mathbf{h}_v^2 - \mathbf{h}_u^2 \mathbf{h}_v^1}, \quad v = \frac{\mathbf{h}_u^4 \mathbf{h}_v^1 - \mathbf{h}_u^1 \mathbf{h}_v^4}{\mathbf{h}_u^1 \mathbf{h}_v^2 - \mathbf{h}_u^2 \mathbf{h}_v^1}, \quad (7)$$

where $\mathbf{h}_u^i, \mathbf{h}_v^i$ are the elements of the 4D homogeneous plane parameters.

Thus far, the Gaussian primitives can be expressed with respect to (u, v) . Suppose $\Sigma^{2D} M = I$, by transforming the 2D covariance matrix Σ^{2D} into the identity matrix I , the probability density function can be rewritten as standardized Gaussian (with mean of zero and deviation of one),

$$\mathcal{G}(\mathbf{x}') = e^{-\frac{1}{2}(M(\mathbf{x}' - \mu'))^\top (M(\mathbf{x}' - \mu'))}, \quad (8)$$

where \mathbf{x}' can be further replaced by (u, v) via some linear transformations as,

$$\mathcal{G}(u, v) = e^{-\frac{1}{2}(u^2 + v^2)}. \quad (9)$$

To further take account of numerical instability introduced by inverse homogeneous transformations of Eq. 4, a lower bounded Gaussian is imposed (Botsch et al. 2005),

$$\hat{\mathcal{G}}(u, v) = \max \left\{ \mathcal{G}(u, v), \mathcal{G}\left(\frac{(u', v') - \mu'}{r}\right) \right\}. \quad (10)$$

When the elliptic disk is projected as the segment line in some cases, a low-pass filter (centered at μ' with radius r) is wielded to guarantee sufficient points passed toward the screen space (the radius is set as $\sqrt{2}/2$ empirically by following (Huang et al. 2024)).

Suppose the opacity of i -th Gaussian primitive is α_i , by considering the alpha-weighted contribution along the ray, the accumulated transmittance is,

$$T_i = \prod_{j=1}^{i-1} \left(1 - \alpha_j \hat{\mathcal{G}}_j(u, v)\right). \quad (11)$$

To this end, the proposed object Gaussian renders both image and depth map of the object. The final color is $c^\alpha = \sum_{i \in \mathcal{N}} T_i \alpha_i \hat{\mathcal{G}}_i(u, v) c_i$, with c_i as the view-dependent appearance represented by spherical harmonics (SH) (Fridovich-Keil et al. 2022; Takikawa et al. 2022). The alpha-blended depth map is formulated via the summation of geometric-aware depth d^{geo} of each Gaussian primitive as,

$$d^\alpha = \sum_{i \in \mathcal{N}} T_i \alpha_i \hat{\mathcal{G}}_i(u, v) \max \{d_i^{\text{geo}} \mid T_i > \sigma\}, \quad (12)$$

where $\sigma = 0.5$ is a threshold deciding whether the rendered depth valid. Noted that the maximum depth along the ray is selected if T_i does not reach the threshold as in (Huang et al. 2024).

Geometric-Consistency under Sparse Views

In the circumstance of extremely sparse view reconstruction, the object Gaussian struggles with over-fitting (Jain, Tancik, and Abbeel 2021; Niemeyer et al. 2022), where the background collapse and floaters are commonly witnessed

even the rendered view deviates marginally from the given one (Xiong et al. 2023). In principle, the effective solutions involve online synthetic view augmentation and geometric-consistent depth supervision. Notably, effective online synthetic view augmentation remarkably reduces the need of a high budget of real images, and the multi-view geometric consistency prevents significant fluctuations on rendered depth.

Synthetic View Warping Given what is at stake, it is intuitive to warp synthetic views online to enrich training samples, which encourages model to adapt from a diverse set of perspectives and brings better generalization capabilities on unseen views.

Owing to lack of ground truth of synthetic views, the proposed alpha-blended depth map plays an essential role in warping process. Specifically, the rendered depth d^α is used to transform the given view into 3D points, which are re-projected as pixels of synthetic views. Formally, the pixel (u'_g, v'_g) of a given view is warped as (u'_w, v'_w) of an unseen view, which is

$$(u'_w, v'_w) = KT \left[d^\alpha K^{-1} (u'_g, v'_g, 1)^\top \right], \quad (13)$$

where K is the camera intrinsic, and the rendered depth d^α serves as the pixel-wise scaling factor to confine the re-projection within a meaningful range.

Moreover, the transformation T from give view to warped view is obtained via perturbations (including rotations and translations) sampled from normal distribution randomly, by making use of tool provided in (Li et al. 2018). The warped pixels are assembled as the ground truth image I_w ,

$$\mathcal{L}_{\text{warp}} = \mathcal{L}_1(\hat{I}, I_w). \quad (14)$$

While the ground truth image I_w and the rendered image \hat{I} of a specific synthetic view establish supervision of the image warping loss via \mathcal{L}_1 .

Depth Supervision under Geometric-Consistent Considering each Gaussian is in tandem with the depth distribution of a certain region in the scene, to concentrate the geometric-aware depth of Gaussian primitives along the ray is beneficial to refine each Gaussian's contribution to overall distribution. Accordingly, the geometric-consistent loss is employed as

$$\mathcal{L}_{\text{geo}} = \sum_{i,j} \omega_i \omega_j |d_i^{\text{geo}} - d_j^{\text{geo}}|, \quad (15)$$

where $\omega_i = T_i \alpha_i \hat{\mathcal{G}}_i(u, v)$ is the blending weight of i -th Gaussian primitive (Huang et al. 2024).

Geometric-consistency Guided Online Pruning Lastly, inspired by (Xiong et al. 2023), an online floaters pruning strategy is implemented by introducing a peak depth,

$$d^{\text{peak}} = d_{\arg \max_i (\omega_i)}^{\text{geo}}. \quad (16)$$

The peak depth (Xiong et al. 2023) is acquired by selecting the Gaussian of highest blending weight, which also is the Gaussian of the highest opacity.

To implement the multi-view geometric-aware depth comparison, the alpha-blending depth d^α and peak depth d^{peak} are compared under each given view. Generally, the alpha-blending depth locates slightly behind the peak depth, the differences result in a candidate region for pruning. While the corresponding opacity α_i of peak depth within the region guides the online floater pruning.

2D-3D Correspondence Generation

The proposed SGPose starts from RGB data alone, without taking advantage of external depth information, yet it effectively renders reliable geometric-aware depth maps. Unlike traditional CAD-free pipelines that heavily rely on geometric initialization from SfM methods such as COLMAP (Schönberger et al. 2016; Schönberger and Frahm 2016), our method handles the random initialization of a cuboid that approximates the bounding box of object. The differentiable optimization of the proposed object Gaussian is expressed as

$$\mathcal{L} = \mathcal{L}_{image} + \mathcal{L}_{warp} + \lambda_1 \mathcal{L}_{geo} + \lambda_2 \mathcal{L}_{normal}. \quad (17)$$

Concretely, \mathcal{L}_{image} is the image rendering loss combining \mathcal{L}_1 with the D-SSIM term from (Kerbl et al. 2023), which is implemented in the given views only. \mathcal{L}_{image} for given views and \mathcal{L}_{warp} for synthetic views follow the identical optimization pipeline, which update the object Gaussian model alternatively. The reasons of implementing such a training strategy are two-folded. Firstly, data from respective views exhibit disparate geometric details, tackling with them independently accommodates the model to the diversified data distributions; Secondly, the stand-alone Gaussian densification and pruning mitigate fluctuations brought by view alternating. λ_1 is set as 10^4 to align up the scale of depth term with the other ones, and $\lambda_2 = 0.005$ for normal loss $\mathcal{L}_{normal} = \sum_i \omega_i (1 - n_i^\top \phi(u, v))$ to facilitate the gradients of depth maps $\phi(u, v)$ in line with normal maps n_i (Huang et al. 2024).

Overall, the geometric-consistent supervision under sparse views reconstructs the desirable object Gaussian. The dense 2D-3D correspondences, generated to fully replace the CAD models, along with synthetic view color images and object masks, serve as the ground truth for a modified GDRNet++ (Liu et al. 2022a) to perform monocular pose regression. Among them, the generation of 2D-3D correspondences and the simulation of realistic occlusions in images are essential for the task.

For dense 2D-3D correspondences, object points are obtained by transforming the rendered depth map into 3D points of camera coordinates via the known camera's intrinsic, and in turn mapping the points to world space via the specific view parameters (rotations and translations). Pixel coordinates are calculated from the rendered object mask. Thus, the 3D points in world space and corresponding pixel coordinates are stack orderly as dense 2D-3D correspondences of any specific view, which is

$$\mathbf{M}_{2D-3D} = \begin{bmatrix} \mathbf{R}_{obj}^\top (d^\alpha K^{-1}(u', v', 1)^\top - \mathbf{t}_{obj}) \\ (u', v')_{mask} \end{bmatrix}, \quad (18)$$

where \mathbf{R}_{obj} and \mathbf{t}_{obj} are the specific view parameters.

Experiments

In this section, extensive experiments are conducted to demonstrate the competitive performance of the proposed SGPose.

Datasets The proposed SGPose is evaluated on two commonly-used datasets, which are LM (Hinterstoisser et al. 2012) and LM-O (Brachmann et al. 2014). LM is a standard benchmark for 6D object pose estimation of textureless objects, which consist of individual sequences of 13 objects of various sizes in the scenes with strong clutters and slight occlusion. Each contains about 1.2k real images with pose annotations. The dataset is split as 15% for training and 85% for testing. LM-O is an extension of LM, from which to annotate one sequence of 8 objects with more severe occlusions of various degrees. For the LM dataset, approximately 1k images with 2D-3D correspondence maps are rendered for each object. For LM-O, which is designed for pose estimation in scenarios with occlusions, 50k images with significant occlusions on transparent backgrounds are rendered and involved in training with a ratio of 10:1, by following the convention of CAD-based methods for a fair comparison (Wang et al. 2021).

Implementation Details Ten real images and corresponding pose annotations are taken as input of the proposed geometric-aware object Gaussian, which are selected from real data in LM. Different selection strategies are conducted, e.g., selecting samples uniformly, randomly, in term of maximum rotation differences and maximum Intersection over Union (IoU). The simple uniform selection is adopted in our method taking account of real-world practice.

The proposed object Gaussian tailors respective optimization strategies to the supervision signals. The number of points of random initialization is set as 4,096. The geometric-consistent loss involves in training at iteration 3,000 and the normal loss is enabled at iteration 7,000 as in (Huang et al. 2024). Ten sparse views are given, and two synthetic views are created online around each give view at iteration 4,999, that is, 30 images involve in training for each object. The synthetic image warping spans 40% of training phase once it is activated, which updates the model alternatively with the image rendering loss. Both image rendering loss and image warping loss are endowed with equal weights, dominating the training process.

Noted that it's a little tricky to conduct pruning effectively in our problem settings. Regions of Interests (RoIs) are remained for training and backgrounds are masked out, it is possible that pruning techniques working well for distant floaters remove part of the foreground objects mistakenly, which is more pronounced when the objects are thin and tall. Thus, the objects phone and driller do not apply pruning empirically.

For the occluded scene generation for LM-O dataset, we utilize pose annotations from PBR (Physically Based Rendering) (Denninger et al. 2023) for image rendering. Each object is rendered onto the image using its respective geometric-aware object Gaussian, with all objects rendered sequentially in a single image. Realizing that individual object Gaussian do not inherently represent occlusions,

Object	DPOD	PVNet	CDPN	GDR-Net	SO-Pose	RLLG	Gen6D [†]	One Pose	One Pose++	GS-Pose	NeRF- Pose	NeRF- Pose [†]	Ours
Views						~200	~200		~180	~180	156	156	10
CAD			w/ CAD						w/o CAD				
Ape	87.73	43.6	67.33	76.29	85.43	52.9	-	11.8	31.2	65.1	89.1	93.1	82.57
Bvise	98.45	99.9	98.74	97.96	99.42	96.5	77.03	92.6	97.3	95.7	99.3	99.6	99.32
Camera	96.07	86.9	92.84	95.29	96.67	87.8	66.67	88.1	88.0	89.4	98.7	98.9	96.18
Can	99.71	95.5	96.56	98.03	98.62	86.8	-	77.2	89.8	97.2	99.1	99.7	99.11
Cat	94.71	79.3	86.63	93.21	95.01	67.3	60.68	47.9	70.4	84.6	97.1	98.1	95.71
Driller	98.80	96.4	95.14	97.72	98.41	88.7	67.39	74.5	92.5	90.7	97.4	98.7	98.91
Duck	86.29	52.6	75.21	80.28	85.73	54.7	40.47	34.2	42.3	72.3	90.3	94.2	85.26
Eggbox	99.91	99.2	99.62	99.53	99.91	94.7	95.7	71.3	99.7	99.2	99.6	99.9	99.81
Glue	96.82	95.7	99.61	98.94	99.61	91.9	87.2	37.5	48.0	88.9	98.1	99.3	99.52
Holep.	86.87	81.9	89.72	91.15	94.77	75.4	-	54.9	69.7	78.6	94.3	96.5	91.91
Iron	100.0	98.9	97.85	98.06	98.67	94.5	-	89.2	97.4	91.7	98.1	97.8	98.47
Lamp	96.84	99.3	97.79	99.14	99.14	96.6	-	87.6	97.8	94.0	97.9	98.7	99.71
Phone	94.69	92.4	90.65	92.35	95.28	89.2	-	60.6	76.0	70.8	96.4	97.3	93.86
Mean	95.15	86.3	91.36	93.69	95.9	82.9	70.73	63.6	76.9	86.0	96.6	97.8	95.41

Table 1: Comparison with state-of-the-arts on the LM w.r.t. the metric of ADD(S)-0.1d. Noted that Gen6D[†] uses a refinement strategy to train on a subset of LM, NeRF-Pose[†] is trained on relative camera pose annotations. The best compared with CAD-free methods are in **bold**, the best compared with CAD-based methods are in *italic bold*.

Object	OnePose	OnePose++	GS-Pose	Ours
Ape	35.2	97.3	97.5	98.67
Bvise	94.4	99.6	98.5	98.64
Camera	96.8	99.6	99.0	99.31
Can	87.4	99.2	97.6	99.51
Cat	77.2	98.7	99.0	99.40
Driller	76.0	93.1	91.9	99.21
Duck	73.0	97.7	97.6	98.59
Eggbox	89.9	98.7	96.9	97.18
Glue	55.1	51.8	96.8	99.03
Holep.	79.1	98.6	98.2	99.33
Iron	92.4	98.9	96.8	98.06
Lamp	88.9	98.8	90.0	95.20
Phone	69.4	94.5	91.1	98.49
Mean	78.1	94.3	96.2	98.51

Table 2: Comparison with state-of-the-arts on the LM w.r.t. the metric of Proj@5pix. Noted that all the other methods use YOLOv5 (Ultralytics 2023) as the object detector and Ours uses YOLOv3 (Redmon and Farhadi 2018). We highlight the best in **bold**.

we overlay the rendered images with the visible masks from PBR data to simulate occlusion scenarios effectively.

The rendered masks, crafted from the proposed geometric-aware object Gaussian, is generated by mapping color images to a binary format. That is, assigning “1” to pixels within the object and “0” to those outside, thus producing a boolean array congruent with the original image’s dimensions. It is possible that incorporating an extra mask loss for supervision could improve the performance of mask rendering in future work.

The 2D bounding boxes for pose estimation are obtained by borrow an off-the-shelf object detector yolov3 (Redmon and Farhadi 2018).

Evaluation Metrics We evaluate our method with the most commonly used metrics including ADD(S)-0.1d and Proj@5pix. ADD(S)-0.1d measures the mean distance between the model points transformed from the estimated pose

and the ground truth. If the percentage of mean distance lies below 10% of the object’s diameter (0.1d), the estimated pose is regarded as correct. For symmetric objects with pose ambiguity, ADD(-S) measures the deviation to the closet model point (Hinterstoisser et al. 2012; Hodan, Barath, and Matas 2020). Proj@5pix computes the mean distance between the projection of 3D model points with given predicted and ground truth object poses. The estimated pose is considered correct if the mean projection distance is less than 5 pixels.

Comparison with State-of-the-Arts

Results on LM The proposed method is compared with the CAD-based methods DPOD (Zakharov, Shugurov, and Ilic 2019), PVNet (Peng et al. 2019), CDPN (Li, Wang, and Ji 2019), GDR-Net (Wang et al. 2021), SO-Pose (Di et al. 2021) and CAD-free methods RLLG (Cai and Reid 2020), Gen6D (Liu et al. 2022b), OnePose (Sun et al. 2022), OnePose++ (He et al. 2022), GS-Pose (Cai, Heikkilä, and Rahtu 2024), and Nerf-Pose (Li et al. 2023) on metric of ADD(S)-0.1d and Proj@5pix. As shown in Tab. 1, even under the setting of sparse view training, our method achieves comparable performance compared to most CAD-free baselines that are trained with more than 100 views, and is on par with the CAD-based methods. Notably, our proposed object Gaussian is trained on only 10 views, whereas Nerf-Pose uses 156 views for OBJ-NeRF training. In brief, the objects where our method outperforms the best CAD-free method (i.e., NeRF-Pose[†]) are highlighted in bold, and where it surpasses the best CAD-based method (i.e., SO-Pose) are in italic bold. Noteworthy, Gen6D[†] is refined on a subset of the LM dataset, and NeRF-Pose[†] is trained on relative camera pose annotations. As show in Tab. 2, SGPose demonstrates an impressive 98.51% average performance using only 10 given views, outperforming all baselines according to the metric of Proj@5pix. Noted that our method uses YOLOv3 (Redmon and Farhadi 2018) as the object detector, while all the others use the more recent YOLOv5 (Ultralyt-

Object	PoseCNN	PVNet	HybridPose	GDR-Net	SO-Pose	GDR-Net	SO-Pose	RLLG	NeRF-Pose	NeRF-Pose [†]	Ours	
CAD	w/ CAD						w/o CAD					
Training	real + syn					real+pbr		real+gen				
Ape	9.6	15.8	20.9	39.3	46.3	46.8	48.4	7.10	46.9	49.7	35.13	
Can	45.2	63.3	75.3	79.2	81.1	90.8	85.8	40.6	86.2	86.4	84.34	
Cat	0.9	16.7	24.9	23.5	18.7	40.5	32.7	15.6	27.1	26.9	23.34	
Driller	41.4	65.7	70.2	71.3	71.3	82.6	77.4	43.9	65.8	66.2	84.68	
Duck	19.6	25.2	27.9	44.4	43.9	46.9	48.9	12.9	29.9	36.9	43.48	
Eggbox	22.0	50.2	52.4	58.2	46.6	54.2	52.4	46.4	24.9	24.4	44.68	
Glue	38.5	49.6	53.8	49.3	63.3	75.8	78.3	51.7	66.3	70.9	69.77	
Holep.	22.1	36.1	54.2	58.7	62.9	60.1	75.3	24.5	46.4	49.8	54.79	
Mean	24.9	40.8	47.5	53.0	54.3	62.2	62.3	30.3	49.2	51.4	55.03	

Table 3: Comparison with state-of-the-arts on the LM-O w.r.t. the metric of ADD(S)-0.1d. “real” is the real data provided by LM-O, “syn” is the blender synthetic data (Li et al. 2018), “pbr” is the physical-based rendering data (Denninger et al. 2023), “gen” is the model generated data. NeRF-Pose[†] is trained on relative camera pose annotations. We highlight the best in **bold**.

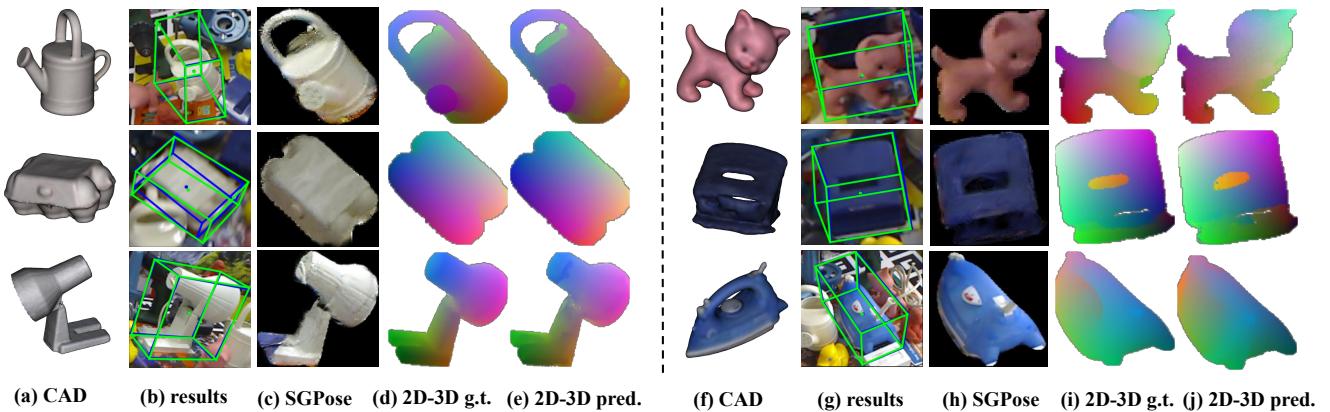


Figure 3: Qualitative results on LM. Column (a) and (f) show the CAD models. Column (b) and (g) illustrate the predicted object poses (green) and ground truth poses (blue). Column (c) and (h) are the rendered images from our object Gaussian. Column (d) and (i) are the generated 2D-3D correspondence maps from our object Gaussian, which are also g.t. of regression network. Column (e) and (j) are the predicted 2D-3D correspondence maps from our regression network.

Object	Individual object	Occluded object
Ape	26.84	35.13
Can	72.83	84.34
Cat	14.57	23.34
Driller	60.71	84.68
Duck	31.15	43.48
Eggbox	19.32	44.68
Glue	38.43	69.77
Holep.	41.4	54.79
Mean	38.15	55.03

Table 4: Comparison of individual object rendering and occluded object rendering on the LM-O w.r.t. the ADD(S)-0.1d.

ics 2023).

Results on LM-O As demonstrated in Tab. 3, our method is compared with state-of-the-arts w.r.t. the metric of average recall (%) of ADD(-S). Among CAD-based methods, “real+pbr” outperforms “real+syn” because “pbr” data (Denninger et al. 2023) incorporate occlusions in object placement, with random textures, materials, and lighting, simulating a more natural environment compared to in-

# of real views	# of synthetic views	Training result
30	0	not converge
33	0	converge
20	20	converge
10	20	converge

Table 5: Ablation study on the effectiveness of synthetic view warping for the object **ape** in the LM.

dividually rendered synthetic data. Given the heavy occlusions typical of the LM-O dataset, training with “pbr” data significantly enhances performance. In our setting, we do not have access to CAD models nor do we leverage “pbr” data. Instead, we render the synthetic images that replicates the occlusion scenarios found in the LM-O dataset, using our proposed object Gaussian. Exemplary, we exceed NeRF-Pose (Li et al. 2023) by 5.83% with 55.03% compared to 49.2%, also rival NeRF-Pose[†], which is trained on relative camera poses instead of ground truth pose annotations, by 3.63%. Impressively, we even slightly outperform the best CAD-based method SO-Pose (Di et al. 2021).

	benchvise		camera		cat		can		duck		glue		Mean	
	w/o.	w.	w/o.	w.	w/o.	w.	w/o.	w.	w/o.	w.	w/o.	w.	w/o.	w.
1mm	0.382	0.377	0.503	0.553	0.417	0.417	0.383	0.410	0.379	0.398	0.332	0.346	0.395	0.409
3mm	0.570	0.584	0.710	0.776	0.656	0.663	0.597	0.644	0.596	0.627	0.545	0.564	0.615	0.625
5mm	0.831	0.857	0.921	0.951	0.936	0.948	0.911	0.929	0.960	0.978	0.853	0.858	0.904	0.904

Table 6: Ablations of online pruning with selected objects on LM dataset.

	can		cat		glue	
	w/o.	w.	w/o.	w.	w/o.	w.
ADD(S)-0.1d	96.85	97.34	51.90	88.12	99.13	99.03
Proj@5pix	98.33	98.43	46.91	99.10	93.63	96.91

Table 7: Pose estimation comparison of selected objects on LM dataset w.r.t. ADD(S)-0.1d and Proj@5pix without and with online pruning.

Ablations

Qualitative comparison of 2D-3D Correspondence The qualitative results of selected objects are presented in Fig. 3, where the transformed 3D bounding boxes are overlaid with the corresponding images. As observed, the predicted poses (green boxes) mostly align with the ground truth (blue boxes). The images are cropped and zoomed into the area of interest for better visualization. The rendered images are exhibited in column (c) and (h), compared to the reference CAD models in column (a), our object Gaussian successfully retains both the silhouette and the details of the objects. The accurate geometric shapes rendered from the object Gaussian ensure the performance of the subsequent pose estimation. Nonetheless, given the inherently challenging nature of sparse view reconstruction, some imperfections in the predicted shapes are also evident.

Training with Occlusions Since LM-O is a more challenging dataset presenting complex occlusions of objects, the integration of synthetic data that captures diverse poses and realistic occlusions is beneficial for enhancing performance. We thus avail the object Gaussian to render such images to enrich training. Quantitatively, as shown in Tab. 4, two different synthetic data are rendered for training, "Occluded object" indicates images containing multiple objects with occlusions, whereas "Individual object" signifies images with a single unoccluded object. We observe that the use of "Occluded object" rendering improves the performance of the proposed SGPose by large margins under all the objects.

For the LM-O dataset, the proposed SGPose generates images and 2D-3D correspondences that demonstrate a diverse range of poses and realistic occlusions, as shown in Fig. 4. In the training process, the synthetic images are integrated at a 10:1 ratio with real images, meaning that for every ten real images, one synthetic image is included. The 2D-3D correspondence maps are projected onto the target object in the images for visualization. Compared to training with individual object rendering, the inclusion of occluded object rendering remarkably enhances the model's performance in complex scenarios where objects have partial visibility.

Effectiveness of Synthetic View Warping As shown in Tab. 5, successful reconstruction of the object ape in LM requires a minimum of 33 images without synthetic view warping. By synthesising 20 novel view alone with 10 given images, the proposed SGPose maintains the performance. This demonstrates the effectiveness of synthetic view warping in reducing the reliance on real images.

Effectiveness of Online Pruning Point cloud accuracy is quantified as the proportion of reconstructed point clouds that fall within a specified distance threshold (e.g., 3mm) relative to the ground truth point clouds, where the vertices of the object meshes serve as the ground truth reference (Sarlin et al. 2023; Schops et al. 2017). The point cloud accuracy is evaluated without and with online pruning for our proposed object Gaussian, following the established protocols in (Sarlin et al. 2023) and (He et al. 2022). The results presented in Tab. 6 demonstrate that the online pruning removes outliers from sparse view object Gaussian reconstruction, resulting in a more accurate and compact representation. Additionally, pose estimation comparison of selected objects w.r.t. ADD(S)-0.1d and Proj@5pix without and with online pruning is presented in Tab. 7. Besides, sparse view reconstruction poses a challenge for object with thin and long geometry, such as lamp and glue. The timely application of online pruning, initiated as divergence threatens, ensures the model to be reconstructed successfully.

Qualitative Results of Synthetic View Warping

The synthetic view for each real image is generated by introducing a controlled amount of noise to the given view, ensuring that the synthetic images retain a realistic and plausible appearance. The perturbation parameters are carefully chosen to keep the object within the camera's field of view. Specifically, the Euler angles for rotation perturbation are sampled from a normal distribution with a standard deviation of 15°, capped at an upper limit of 45°. The translation perturbation along each axis is independently sampled from a normal distribution with standard deviations of 0.01 m for the x and y axes, and 0.05 m for the z-axis, respectively. Fig. 5 displays the qualitative results. Column (a) and (e) presents the ground truth of given views, while column (b) and (f) shows the corresponding rendered images from SGPose. Columns (c) and (g) illustrate the ground truth of synthetic views, and columns (d) and (h) exhibit the rendered synthetic images, respectively. The rendered results are obtained at the iteration 30k upon completion of the training.

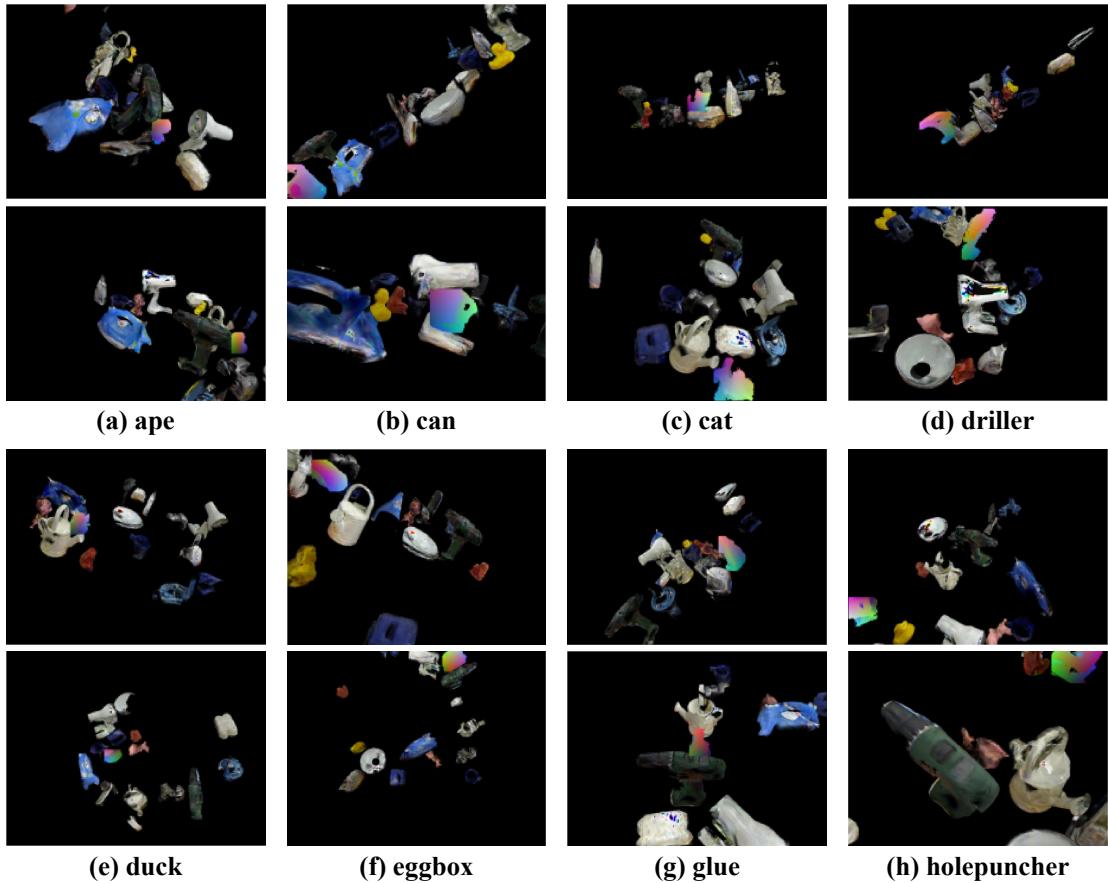


Figure 4: Qualitative results for each object of synthetic data for the LM-O dataset, where the 2D-3D correspondences are projected onto the target object for visualization.



Figure 5: Qualitative results for selected synthetic views. Column (a) and (e) display ground truth of given views; (b) and (f) show SGPose rendered images of given views. Synthetic view ground truths are in (c) and (g), with their corresponding rendered images in (d) and (h). Synthetic views are generated by applying rotation perturbations of up to $\pm 15^\circ$ and translation perturbations of ± 0.01 m along the x and y axes, and ± 0.05 m along the z-axis to the given views.

Implementation and Runtime Analysis

The experiments are conduct on an platform with an Intel(R) Xeon(R) Gold 5220R 2.20GHz CPU and Nvidia RTX3090 GPUs of 24GB Memory. Given ten 640×480 images as input, the object Gaussian costs about 10 minutes to reconstruct one object and real-time renders image, mask, and 2D-3D correspondence. An ImageNet (Deng et al. 2009) pre-trained ConvNeXt (Liu et al. 2022c) network is leveraged as the backbone of our pose regression network, for a 640×480 image, the proposed SGPose takes about 24ms for inference.

Limitations

In future work, we plan to reduce the training time to enable portable online reconstruction and pose estimation, thereby facilitating real-time, end-to-end pose estimation suitable for real-world applications.

Conclusion

The proposed SGPose presents a monocular object pose estimation framework, effectively addressing the limitations of traditional methods that rely on CAD models. By introducing a novel approach that requires as few as ten reference views, the derived geometric-aware depth guides the object-centric Gaussian model to perform synthetic view warping and online pruning effectively, showcasing its robustness and applicability in real-world scenarios under sparse view constraints. The occlusion data rendered from the proposed object Gaussian substantially enhances the performance of post estimation, set our SGPose a state-of-the-art on the Occlusion LM-O dataset.

References

- Ahmadyan, A.; Zhang, L.; Ablavatski, A.; Wei, J.; and Grundmann, M. 2021. Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 7822–7831.
- Azad, P.; Asfour, T.; and Dillmann, R. 2007. Stereo-based 6d object localization for grasping with humanoid robot systems. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 919–924. IEEE.
- Botsch, M.; Hornung, A.; Zwicker, M.; and Kobbelt, L. 2005. High-quality surface splatting on today’s GPUs. In *Proceedings Eurographics/IEEE VGTC Symposium Point-Based Graphics*, 2005., 17–141. IEEE.
- Brachmann, E.; Krull, A.; Michel, F.; Gumhold, S.; Shotton, J.; and Rother, C. 2014. Learning 6d object pose estimation using 3d object coordinates. In *European conference on computer vision*, 536–551. Springer.
- Cai, D.; Heikkilä, J.; and Rahtu, E. 2024. Gs-pose: Cascaded framework for generalizable segmentation-based 6d object pose estimation. *arXiv preprint arXiv:2403.10683*.
- Cai, M.; and Reid, I. 2020. Reconstruct locally, localize globally: A model free method for object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3153–3163.
- Chen, X.; Dong, Z.; Song, J.; Geiger, A.; and Hilliges, O. 2020. Category level object pose estimation via neural analysis-by-synthesis. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*, 139–156. Springer.
- Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.
- Denninger, M.; Winkelbauer, D.; Sundermeyer, M.; Boerdijk, W.; Knauer, M.; Strobl, K. H.; Humt, M.; and Triebel, R. 2023. BlenderProc2: A Procedural Pipeline for Photorealistic Rendering. *Journal of Open Source Software*, 8(82): 4901.
- Di, Y.; Manhardt, F.; Wang, G.; Ji, X.; Navab, N.; and Tombari, F. 2021. So-pose: Exploiting self-occlusion for direct 6d pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 12396–12405.
- Fridovich-Keil, S.; Yu, A.; Tancik, M.; Chen, Q.; Recht, B.; and Kanazawa, A. 2022. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5501–5510.
- He, X.; Sun, J.; Wang, Y.; Huang, D.; Bao, H.; and Zhou, X. 2022. Onepose++: Keypoint-free one-shot object pose estimation without CAD models. *Advances in Neural Information Processing Systems*, 35: 35103–35115.
- Hinterstoisser, S.; Lepetit, V.; Ilic, S.; Holzer, S.; Bradski, G.; Konolige, K.; and Navab, N. 2012. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Asian conference on computer vision*, 548–562. Springer.
- Hodan, T.; Barath, D.; and Matas, J. 2020. Epos: Estimating 6d pose of objects with symmetries. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11703–11712.
- Huang, B.; Yu, Z.; Chen, A.; Geiger, A.; and Gao, S. 2024. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 Conference Papers*, 1–11.
- Jain, A.; Tancik, M.; and Abbeel, P. 2021. Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5885–5894.
- Kehl, W.; Manhardt, F.; Tombari, F.; Ilic, S.; and Navab, N. 2017. Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *Proceedings of the IEEE international conference on computer vision*, 1521–1529.
- Kerbl, B.; Kopanas, G.; Leimkühler, T.; and Drettakis, G. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Trans. Graph.*, 42(4): 139–1.
- Kopanas, G.; Philip, J.; Leimkühler, T.; and Drettakis, G. 2021. Point-Based Neural Rendering with Per-View Optimization. In *Computer Graphics Forum*, volume 40, 29–43. Wiley Online Library.
- Labbé, Y.; Carpentier, J.; Aubry, M.; and Sivic, J. 2020. Cosopose: 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, 574–591. Springer.
- Lee, T.; Lee, B.-U.; Kim, M.; and Kweon, I. S. 2021. Category-level metric scale object shape and pose estimation. *IEEE Robotics and Automation Letters*, 6(4): 8575–8582.
- Lepetit, V.; Moreno-Noguer, F.; and Fua, P. 2009. Epnp: An accurate $O(n)$ solution to the pnp problem. *International journal of computer vision*, 81(2): 155–166.
- Li, F.; Vutukur, S. R.; Yu, H.; Shugurov, I.; Busam, B.; Yang, S.; and Ilic, S. 2023. Nerf-pose: A first-reconstruct-then-regress approach for weakly-supervised 6d object pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2123–2133.
- Li, Y.; Wang, G.; Ji, X.; Xiang, Y.; and Fox, D. 2018. Deepim: Deep iterative matching for 6d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 683–698.
- Li, Z.; Wang, G.; and Ji, X. 2019. Cdpn: Coordinates-based disentangled pose network for real-time rgb-based 6-dof object pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7678–7687.
- Liu, X.; Zhang, R.; Zhang, C.; Fu, B.; Tang, J.; Liang, X.; Tang, J.; Cheng, X.; Zhang, Y.; Wang, G.; and Ji, X. 2022a. GDRNPP. <https://github.com/shanice-l/gdrnpp.bop2022>.
- Liu, Y.; Wen, Y.; Peng, S.; Lin, C.; Long, X.; Komura, T.; and Wang, W. 2022b. Gen6d: Generalizable model-free 6-dof object pose estimation from rgb images. In *European Conference on Computer Vision*, 298–315. Springer.
- Liu, Z.; Mao, H.; Wu, C.-Y.; Feichtenhofer, C.; Darrell, T.; and Xie, S. 2022c. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11976–11986.
- Manhardt, F.; Kehl, W.; and Gaidon, A. 2019. Roi-10d: Monocular lifting of 2d detection to 6d pose and metric shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2069–2078.
- Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1): 99–106.
- Niemeyer, M.; Barron, J. T.; Mildenhall, B.; Sajjadi, M. S.; Geiger, A.; and Radwan, N. 2022. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5480–5490.
- Oberweger, M.; Rad, M.; and Lepetit, V. 2018. Making deep heatmaps robust to partial occlusions for 3d object pose estimation. In *Proceedings of the European conference on computer vision (ECCV)*, 119–134.
- Park, K.; Mousavian, A.; Xiang, Y.; and Fox, D. 2020. 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*, 10710–10719.
- Park, K.; Patten, T.; and Vincze, M. 2019. Pix2pose: Pixel-wise coordinate regression of objects for 6d pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 7668–7677.
- Pavlakos, G.; Zhou, X.; Chan, A.; Derpanis, K. G.; and Daniilidis, K. 2017. 6-dof object pose from semantic keypoints. In *2017 IEEE international conference on robotics and automation (ICRA)*, 2011–2018. IEEE.
- Peng, S.; Liu, Y.; Huang, Q.; Zhou, X.; and Bao, H. 2019. PVNet: Pixel-wise voting network for 6DoF object pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(8).
- Qi, C. R.; Liu, W.; Wu, C.; Su, H.; and Guibas, L. J. 2018. Frustum pointnets for 3d object detection from rgbd data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 918–927.
- Redmon, J.; and Farhadi, A. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
- Sarlin, P.-E.; Lindenberger, P.; Larsson, V.; and Pollefeys, M. 2023. Pixel-perfect structure-from-motion with feature-metric refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Schönberger, J. L.; and Frahm, J.-M. 2016. Structure-from-Motion Revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Schönberger, J. L.; Zheng, E.; Pollefeys, M.; and Frahm, J.-M. 2016. Pixelwise View Selection for Unstructured Multi-View Stereo. In *European Conference on Computer Vision (ECCV)*.
- Schops, T.; Schonberger, J. L.; Galliani, S.; Sattler, T.; Schindler, K.; Pollefeys, M.; and Geiger, A. 2017. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3260–3269.
- Sigg, C.; Weyrich, T.; Botsch, M.; and Gross, M. H. 2006. GPU-based ray-casting of quadratic surfaces. In *PBG@ SIGGRAPH*, 59–65.
- Sun, J.; Wang, Z.; Zhang, S.; He, X.; Zhao, H.; Zhang, G.; and Zhou, X. 2022. Onepose: One-shot object pose estimation without cad models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6825–6834.
- Takikawa, T.; Evans, A.; Tremblay, J.; Müller, T.; McGuire, M.; Jacobson, A.; and Fidler, S. 2022. Variable bitrate neural fields. In *ACM SIGGRAPH 2022 Conference Proceedings*, 1–9.
- Tan, D. J.; Tombari, F.; and Navab, N. 2018. Real-time accurate 3d head tracking and pose estimation with consumer rgbd cameras. *International Journal of Computer Vision*, 126: 158–183.
- Tian, M.; Ang, M. H.; and Lee, G. H. 2020. Shape prior deformation for categorical 6d object pose and size estimation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, 530–546. Springer.

Ultralytics. 2023. Yolov5: Real-time object detection. https://github.com/shanice-l/gdrnpp_bop2022.

Vince, J. 2008. *Geometric algebra for computer graphics*. Springer Science & Business Media.

Wang, G.; Manhardt, F.; Tombari, F.; and Ji, X. 2021. Gdrnet: Geometry-guided direct regression network for monocular 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16611–16621.

Wang, H.; Sridhar, S.; Huang, J.; Valentin, J.; Song, S.; and Guibas, L. J. 2019. Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2642–2651.

Wang, J.; Chen, K.; and Dou, Q. 2021. Category-level 6d object pose estimation via cascaded relation and recurrent reconstruction networks. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4807–4814. IEEE.

Weyrich, T.; Heinze, S.; Aila, T.; Fasnacht, D. B.; Oetiker, S.; Botsch, M.; Flraig, C.; Mall, S.; Rohrer, K.; Felber, N.; et al. 2007. A hardware architecture for surface splatting. *ACM Transactions on Graphics (TOG)*, 26(3): 90–es.

Xiang, Y.; Schmidt, T.; Narayanan, V.; and Fox, D. 2017. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*.

Xiong, H.; Muttukuru, S.; Upadhyay, R.; Chari, P.; and Kadambi, A. 2023. Sparsegs: Real-time 360 ° sparse view synthesis using gaussian splatting. *arXiv preprint arXiv:2312.00206*.

Zakharov, S.; Shugurov, I.; and Ilic, S. 2019. Dpod: 6d pose object detector and refiner. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1941–1950.

Zwicker, M.; Pfister, H.; Van Baar, J.; and Gross, M. 2001. EWA volume splatting. In *Proceedings Visualization, 2001. VIS'01.*, 29–538. IEEE.

Zwicker, M.; Rasanen, J.; Botsch, M.; Dachsbaecher, C.; and Pauly, M. 2004. Perspective accurate splatting. In *Proceedings-Graphics Interface*, 247–254.