

iGaussian: Real-Time Camera Pose Estimation via Feed-Forward 3D Gaussian Splatting Inversion

Hao Wang*, Linqing Zhao*, Xiuwei Xu, Jiwen Lu, Haibin Yan†

Abstract—Recent trends in SLAM and visual navigation have embraced 3D Gaussians as the preferred scene representation, highlighting the importance of estimating camera poses from a single image using a pre-built Gaussian model. However, existing approaches typically rely on an iterative *render-compare-refine* loop, where candidate views are first rendered using NeRF or Gaussian Splatting, then compared against the target image, and finally, discrepancies are used to update the pose. This multi-round process incurs significant computational overhead, hindering real-time performance in robotics. In this paper, we propose iGaussian, a two-stage feed-forward framework that achieves real-time camera pose estimation through direct 3D Gaussian inversion. Our method first regresses a coarse 6DoF pose using a Gaussian Scene Prior-based Pose Regression Network with spatial uniform sampling and guided attention mechanisms, then refines it through feature matching and multi-model fusion. The key contribution lies in our cross-correlation module that aligns image embeddings with 3D Gaussian attributes without differentiable rendering, coupled with a Weighted Multiview Predictor that fuses features from Multiple strategically sampled viewpoints. Experimental results on the NeRF Synthetic, Mip-NeRF 360, and T&T+DB datasets demonstrate a significant performance improvement over previous methods, reducing median rotation errors to 0.2° while achieving 2.87 FPS tracking on mobile robots, which is an impressive 10× speedup compared to optimization-based approaches.

I. INTRODUCTION

Estimating camera poses from a single image using a pre-built 3D scene representation is a critical capability for applications such as robotic navigation and augmented reality (AR), where real-time localization in unknown environments is paramount. This task involves inferring the 6-DoF camera pose (rotation and translation) of an observed image relative to a pre-optimized 3D Gaussian splatting model, which is a lightweight, differentiable representation that enables efficient scene rendering. Such a capability eliminates the need for repeated scene reconstruction during deployment, allowing robots or AR devices to localize instantly in environments where prior mapping has been completed.

The fundamental challenge lies in bridging the gap between the expressive power of 3D Gaussian representations and the need for feed-forward pose inference. Existing methods typically rely on iterative optimization frameworks that render candidate views,

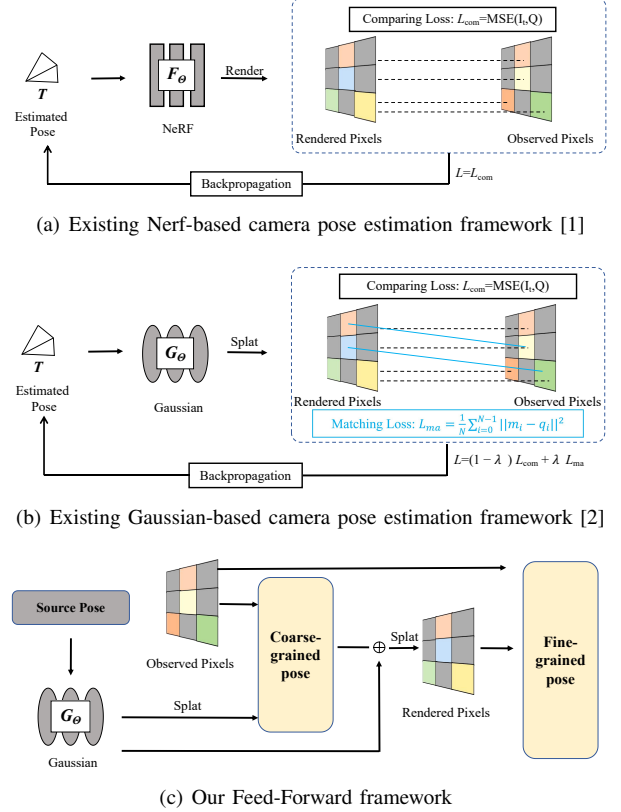


Fig. 1: Comparison of existing pose estimation methods based on (a) NeRF [3], (b) 3DGS [4], and (c) our method. Both (a) and (b) rely on multiple “render-compare-refine” iterations for optimization, whereas our approach follows a feed-forward paradigm.

compare them against the target image, and refine poses through gradient-based updates, a computationally expensive process incompatible with real-time requirements. While Gaussians enable photorealistic rendering, their unstructured nature complicates direct geometric reasoning, and iterative refinement introduces latency that scales poorly with scene complexity.

While Neural Radiance Fields (NeRF) [3] and subsequent works [5] achieve accurate scene reconstruction via photometric optimization, their reliance on volumetric rendering and iterative pose refinement results in impractical computational costs. Recent 3D Gaussian-based SLAM systems [6] address rendering efficiency through splatting but inherit two core limitations: (1) Persistent dependency on slow “render-compare-refine” loops for joint Gaussian-camera optimization, and (2) Requirement for depth sensors to bootstrap geometry, restricting deployment to RGB-

*Equal contribution. †Corresponding author.

Hao Wang and Haibin Yan are with the School of Intelligent Engineering and Automation, Beijing University of Posts and Telecommunications, Beijing 100876, China. Email: {2673439694, eyanhaibin}@bupt.edu.cn.

Linqing Zhao and Xiuwei Xu are with the Department of Automation, Tsinghua University, Beijing, 100084, China. Email: zhaolinqing@mail.tsinghua.edu.cn; xxw21@mails.tsinghua.edu.cn.

only platforms. Optimization-driven methods like iNeRF [1] and iComMa [2] further exemplify this trade-off—while achieving sub-degree pose accuracy, they demand hundreds of iterations per frame, rendering them unsuitable for real-time robotics. These constraints highlight an unresolved tension between accuracy and efficiency in pose estimation.

In this paper, we aim to bridge through a feed-forward inversion paradigm that directly maps image features to camera poses, bypassing both iterative optimization and depth dependency. We first introduce a spherical sampling strategy to enable efficient camera pose estimation from pre-trained 3D Gaussian scene models. At its core, the method uniformly samples multiple candidate viewpoints on a sphere encompassing the target scene, generating synthetic reference images through Gaussian splatting. These reference images, paired with the observed target image, are fed into a pose regression network that directly predicts a coarse 6DoF camera pose in a single forward pass. This network leverages cross-view geometric relationships through an attention mechanism, focusing on spatially consistent features across sampled viewpoints to resolve pose ambiguities. To refine the initial prediction, we employ a hybrid optimization stage: A synthetic image is rendered using the coarse pose guides feature matching with the target image, while a vision transformer resolves residual pose errors by analyzing relative spatial transformations. Crucially, the entire pipeline operates without iterative optimization or depth sensors, achieving real-time performance while retaining the accuracy of SOTA Gaussian-based SLAM systems. Experiments show that our method outperforms optimization-driven baselines both in terms of performance and speed, enabling reliable deployment on resource-constrained robotic platforms.

II. RELATED WORK

Learning-based Camera Pose Estimation. Recent advances in deep learning have revolutionized camera pose estimation by replacing traditional geometric pipelines with data-driven frameworks. While early attempts concatenated multi-view image features for direct pose regression [7], [8], [9], these suffered from limited generalization and scale ambiguity. Subsequent advances introduced correlation volumes [10] and transformer architectures [11], [12] to enable more sophisticated cross-view feature interaction. Recent methods integrate geometric priors into neural networks through architectural innovations - 8-Point ViT [12] embeds fundamental matrix constraints into vision transformers, while RPNNet [13] combines absolute pose regression with geometric verification. Parallel efforts focus on reconstructing planar 3D structures from minimal wide-baseline image pairs [14], [15], demonstrating the power of geometric-aware learning. Our approach draws inspiration from FAR [16], which establishes a hybrid framework combining learned correspondence priors with robust solver optimization to achieve SOTA accuracy. Feature matching-based pose estimation remains a fundamental problem in computer vision, with classical approaches relying on handcrafted descriptors (e.g., SIFT [17]) combined with

RANSAC [18] and the 8-point algorithm [19], [20]. While these methods demonstrate robustness to noise, they often struggle in views with extreme viewpoint variations or low-texture surfaces. Recent advancements have introduced learning-based frameworks to overcome these limitations. Methods such as SuperGlue [21], LoFTR [22], and MatchFormer [23] leverage deep neural networks to enhance feature correspondence quality, improving robustness across challenging conditions. Additionally, approaches like DeepIM [24] and MOPED [25] have enabled real-time 6D pose estimation by integrating learning-based refinement with geometric reasoning. Beyond image-based matching, several methods focus on aligning images with target point clouds or 3D models [26], [27], [28], achieving good performance in scenarios requiring accurate spatial alignment.

NeRF and Gaussian Splatting for Pose Estimation. Implicit neural representations like NeRF and 3DGS have revolutionized simultaneous localization and mapping (SLAM) by unifying scene reconstruction and camera tracking within a differentiable framework. NeRF-based SLAM systems [29], [30], [31] and iNeRF [1] jointly optimize camera poses with radiance fields through photometric consistency in volume rendering. In contrast, 3DGS [4] parameterizes scenes as anisotropic 3D Gaussians with explicit spatial control, enabling efficient rasterization via splatting-based rendering and dynamic scene adaptation through Gaussian attribute optimization. Recent advancements like Gaussian Splatting SLAM [32] and SemGauss-SLAM [33] achieve real-time free-viewpoint streaming by training Gaussians on the fly, reducing computational latency by an order of magnitude compared to NeRF-based systems. These approaches demonstrate enhanced scalability and geometric fidelity over traditional discrete representations [34], particularly in preserving high-frequency details and supporting dynamic object handling through techniques. However, challenges persist in textureless region reconstruction and extreme occlusion scenarios due to the dependency on photometric cues. Our framework in Fig. 1 addresses these limitations, and implements a coarse-to-fine Gaussian selection strategy to prioritize geometrically reliable regions during optimization. Our methodology similarly adopts 3DGS as a scene prior to guide pose estimation, akin to iComMa [2], leveraging its powerful rendering capabilities to achieve robust online operational performance.

III. APPROACH

We propose iGaussian, which aims to predict the camera pose of the observed viewpoint from a 3D Gaussian scene representation. Assuming that the 3D Gaussian [4] model of the scene or object is represented by parameters Θ , and the camera’s intrinsic parameters are known, but the camera pose T of the target image I remains unknown. Our goal is to solve for the camera pose T_{fine} of the observed image I . Our method can infer the camera pose T_{fine} directly from

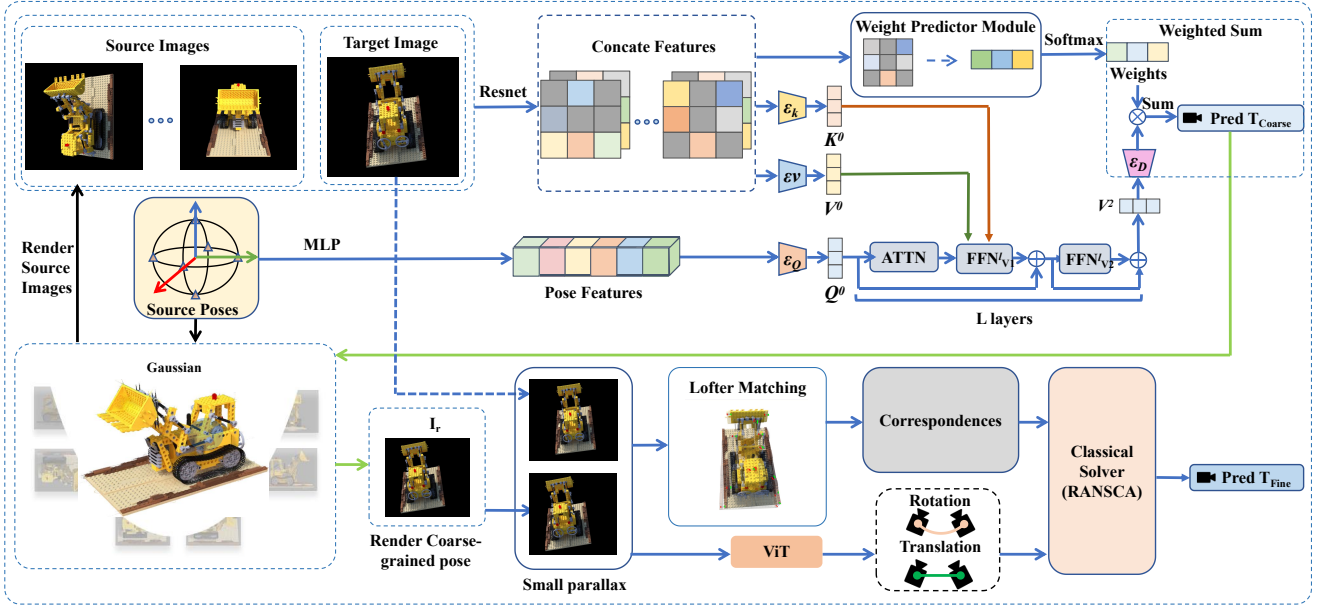


Fig. 2: Overview of iGaussian pipeline. Our approach estimates the camera pose T_{fine} of an observed image I using a two-stage pipeline. First, a Pose Attention Network predicts a coarse 6DoF pose (R, t) from the target image and generates a reference view using a 3D Gaussian representation. Then, a matching and solver module refines the pose by computing the relative transformation between the observed and rendered images. A Transformer-based ViT predicts translation scale and learned correspondence estimation with geometric constraints to enhance accuracy. The framework eliminates iterative rendering, ensuring efficient and precise pose regression.

the input image and scene parameters Θ , independent of reference images.

iGaussian combines the strengths of correspondence estimation and end-to-end pose estimation, leveraging a 3D Gaussian model as a prior for precise pose estimation. As shown in Fig. 2, our method operates in two stages. First, the Rendering-Based Coarse-Grained Pose Estimation network regresses a coarse 6DoF pose (R, t) from the observed image I and renders a reference view I_r using a 3D Gaussian rendering pipeline. Second, a matching and solver module computes the relative pose ΔT between I and I_r , guided by a Transformer-based ViT [12], [35] that predicts the Scale. This architecture eliminates the need for iterative “compare-render-compare” steps, relying instead on a single rendering and high-quality correspondences for efficient and accurate pose estimation.

A. Rendering-Based Coarse-Grained Pose Estimation

Given a target image I and a 3D Gaussian representation Θ , our network estimates the absolute pose by leveraging spatial constraints and source image pose information to map input to predicted pose. Unlike traditional view synthesis networks, our approach focuses on learning 3D positional relationships rather than pixel-level transformations, enabling precise 6DoF pose prediction and effective cross-viewpoint pose distribution modeling for accurate camera pose regression.

a) *Spatial Uniform Sampling Strategy*: Since our input merely encompasses the target image I and the 3D Gaussian representation Θ of an object or scene, we developed a

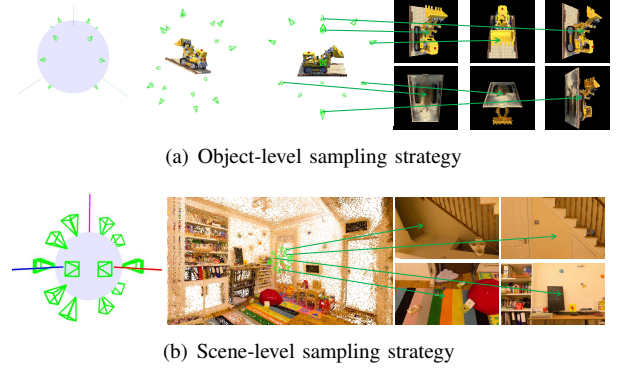


Fig. 3: Object-level and scene-level sampling strategies. (a) and (b) illustrate different sampling strategies for objects and scenes, respectively. In both cases, cameras are uniformly distributed on a spherical surface. However, for objects, the camera viewpoints are always directed toward the object’s center, ensuring comprehensive coverage. In contrast, for scenes, the camera viewpoints consistently face away from the scene’s center, capturing a broader environmental context.

spatial uniform sampling strategy to sample source viewpoint cameras from the Gaussian model as a reference. To ensure that the sampled source viewpoints uniformly cover the entire 3D scene, we constructed a viewpoint distribution strategy based on the 3D Gaussian model and spherical coordinate system in Fig. 3. Specifically, we define a spherical trajectory centered at the coordinate origin with a fixed radius R , allowing cameras to be uniformly distributed on the sphere. By adjusting the polar angle Φ and azimuthal angle ψ , we achieve uniform sampling. In the spherical coordinate

system, the 3D position $p = (x, y, z)$ of the camera is calculated as:

$$\begin{cases} x = R \cdot \sin(\Phi) \cdot \cos(\psi) \\ y = R \cdot \sin(\Phi) \cdot \sin(\psi) \\ z = R \cdot \cos(\Phi), \end{cases} \quad (1)$$

where R is the spherical radius, θ is the polar angle, and ψ is the azimuthal angle. To ensure each sampled camera viewpoint points toward the object's center (the origin), we generate the view matrix \mathbf{M} using the *look_at* function [36]:

$$\mathbf{M} = \text{look_at}(p, \text{target}, up), \quad (2)$$

where p is the camera position, *target* is the focal point, and $up = (0, 1, 0)$ or $(0, -1, 0)$ defines the upward direction. By controlling the sampling points of θ and ψ , we generate camera positions and compute their corresponding view matrices. These matrices are used to render source images $\{I_s^{(i)}\}_{i=1}^N$, forming a complete source viewpoint set \mathcal{S} . The target image I , along with the N source images, is input into the network for pose regression:

$$T = f_\theta(I, \{I_s^{(i)}\}_{i=1}^N; \Theta). \quad (3)$$

This approach ensures that the camera trajectory is independent of scene complexity, allowing the same strategy to be applied to scenes of varying scales and complexities without requiring additional depth information estimation.

b) Attention-based Spatial Transformation Learning:

In the absence of appropriate inductive biases, networks trained solely on images may struggle to generalize, particularly in tasks involving direct learning of geometric transformations rather than pixel-level mapping. Directly regressing pose from image features without explicit spatial correspondences often leads to unstable or imprecise predictions. To address this, we propose a method that guides self-attention for spatial transformation learning, departing from traditional appearance-based approaches [37]. Specifically, we encode the pose of source images as query vectors Q^i and direct it to focus on the most critical components for pose estimation within the concatenated features K^i and V^i formally:

$$K^i, V^i = \text{concat}(\phi_r(I), \phi_r(x_s^i)), \quad (4)$$

$$Q^i = \text{MLP}(\text{Pose}(x_s^i)), \quad (5)$$

$$\text{Attention}(Q^i, K^i, V^i) = \text{softmax}\left(\frac{Q^i(K^i)^T}{\sqrt{d_k}}\right)V^i, \quad (6)$$

where ϕ_r is the image feature extractor (a pre-trained ResNet in this work), and $\text{Pose}(x_s^i)$ represents the pose information of source images. The query vector Q^i is projected into a 128-dimensional embedding via an MLP. The attention output is refined through feed-forward networks (FFNs) and decoded to the image space:

$$V^{i1} = \text{FFN}_1(\text{Attention}(Q^i, K^i, V^i)) + V^i, \quad (7)$$

$$V^{i2} = \text{FFN}_2(V^{i1}), \quad (8)$$

$$\hat{T}^i = \xi_D(V^{i2}). \quad (9)$$

By concatenating target and source image features as K^i and V^i , we enhance cross-view geometric information, improving spatial alignment and providing stable geometric cues. The MLP-projected query vectors Q^i explicitly guide attention to regions relevant to target viewpoint changes, ensuring stable attention distribution and mitigating feature misalignment caused by viewpoint variations.

c) *Multi-viewpoint Feature Fusion and Pose Regression:* Given a set of source images $\{x_s^i\}_{i=1}^N$ and a target image I , we first extract and concatenate their features and then predict multiple candidate poses using a pose regression network:

$$\hat{T}^i = f(x_s^i, I; \theta), \quad (10)$$

where x_s^i and I represent the source and target images, respectively, and $f(\cdot)$ is the pose regression network predicting the camera pose \hat{T}^i . To address the imbalance in information provided by different source viewpoints, we designed the Weight Predictor Module (WPM), which assigns weights to each predicted pose based on feature information. The weights w^i are predicted using an MLP:

$$w^i = g(f(x_s^i, I)), \quad (11)$$

where $g(\cdot)$ is the weight prediction network. The weights are normalized via Softmax to form a probability distribution:

$$w^i = \frac{\exp(w^i)}{\sum_{j=1}^N \exp(w^j)} g(f(x_s^i, I)), \quad (12)$$

The normalized weights w^i indicate the importance of each x_s^i for candidate pose prediction. The final predicted pose T_{coarse} is computed as a weighted average:

$$T_{coarse} = \sum_{i=1}^N w^i \cdot \hat{T}^i. \quad (13)$$

Through spherical uniform sampling, we ensure comprehensive coverage of source viewpoints, reducing ambiguity in pose estimation. The WPM enhances prediction accuracy by assigning higher weights to more reliable viewpoints.

B. Correspondence-Based Pose Optimization

After predicting an initial pose T_{coarse} using a Gaussian scene prior-based pose regression network and rendering an image I_r via the 3D Gaussian rendering pipeline, we obtain a rendered image close to the target image I . Although the visual difference between I_r and I is small, further refinement is necessary. In our method, relative pose optimization is a critical step to enhance the accuracy of the initial pose prediction T_{coarse} , aligning I and I_r more precisely. This process integrates the translation scale predicted by ViT with traditional feature matching (LoFTR [22] and RANSAC [38]), achieving accurate optimization of the predicted pose T_{coarse} .

a) *Feature Point Matching and Relative Pose Calculation.*: In correspondence-based pose estimation, methods like RANSAC and its variants are commonly used. These methods randomly sample minimal point sets and fit models using n-point algorithms, with Sampson error as the inlier threshold. Given a set of 2D correspondences $M=\{(p, q)\}$, the scoring function for hypothesis H [39] is:

$$\text{score}(H) = \sum_{(p,q) \in M} 1(E(p, q|H) < \sigma), \quad (14)$$

where $E(p, q|H)$ is the Sampson error, the model with the highest score (most inliers) is selected. This sampling process repeats N times until convergence. Under the initial pose prediction T_{coarse} , feature point matching computes the corresponding feature set $\{(p_i, q_i)\}$ between the source image I_r and target image I where p_i and q_i are matching points. The close viewpoints yield abundant matches, enabling precise relative pose estimation, with RANSAC solving for the relative pose ΔT :

$$\Delta T = \text{RANSAC}(p_i, q_i). \quad (15)$$

The resulting relative pose ΔT includes the rotation matrix ΔR and translation vector Δt , but lacks translation scale. Traditional geometric methods like RANSAC and the 8-Point Algorithm excel in small-disparity views, where high viewpoint overlap ensures accurate results. In our approach, the Gaussian scene prior-based pose regression network provides an accurate initial pose T_{coarse} and rendered image I_r , enabling RANSAC to compute with high precision.

b) *Translation Scale Correction.*: Since the relative pose ΔT obtained via RANSAC lacks translation scale, we employ ViT to predict the relative pose T_{vit} with translation scale between the target and rendered images. This enhances the solver’s robustness by integrating learning-based predictions. The translation scale $|t_{vit}|$ is used to correct ΔT :

$$|t_{fine}| = |\Delta t| \times |t_{vit}|. \quad (16)$$

The corrected T_{fine} provides a more accurate representation of the relative pose between the source and target images. Traditional geometric methods excel in small-disparity views but struggle with translation scale and complex scenes. Deep learning methods like ViT adapt better to complex scenes but rely on training data and architecture. By combining their strengths, our approach achieves multi-stage pose optimization, enhancing accuracy and stability.

C. Losses

We define a pose loss function that combines rotation and translation errors.

a) *Rotation Error.*: The rotation error is computed as the angular difference between the predicted and ground truth rotation matrices. Given two unit quaternions q_{fine} and q representing the predicted and ground truth rotations, respectively, the rotation error L_{rot} is defined as:

$$L_{rot} = 2 \cdot \arccos(|q_{fine} \cdot q|) \cdot \frac{180}{\pi}, \quad (17)$$

where $|q_{fine} \cdot q|$ denotes the dot product of the two quaternions.

b) *Translation Error.*: The translation error L_{trans} measures the Euclidean distance between the predicted translation vector t_{fine} and the ground truth translation vector t :

$$L_{trans} = \|t_{fine} - t\|_2, \quad (18)$$

where $\|\cdot\|_2$ denotes the L_2 -norm.

c) *Overall Loss Function.*: The overall pose loss L_{pose} is a weighted combination of the rotation and translation errors:

$$L_{pose} = \lambda_r \cdot L_{rot} + \lambda_t \cdot L_{trans}, \quad (19)$$

where λ_r and λ_t are hyperparameters that control the relative importance of the rotation and translation errors, respectively. In our implementation, we set $\lambda_r = 1.0$, $\lambda_t = 30$.

IV. EXPERIMENT

We conducted a series of experiments to evaluate the performance of our proposed method in the 6DoF pose estimation task and compared it with SOTA approaches. To comprehensively analyze the model’s adaptability across different scenarios or object, we tested it on multiple benchmark datasets. First, we test it on 8 objects from NeRF’s synthetic dataset [3], which primarily consist of single-object scenes, providing a controlled environment for performance evaluation. Additionally, we employed 360-degree unbounded scene datasets from Mip-NeRF360 [40], which offer a more comprehensive reflection of the model’s capabilities in large-scale and complex environments. To further verify the model’s generalization ability in real-world scenarios, we introduced the T&T + DB [41] dataset, which represents typical 360-degree scenes and includes various real-world environmental variations. We also conducted ablation studies to assess the contribution of different modules to the final performance. In all experiments, we uniformly used 16 source images as input, selected based on a specific sampling strategy, where the elevation angle Φ was set to $\pm[30^\circ, 60^\circ]$ and the azimuth angle ψ was sampled at $[0^\circ, 90^\circ, 180^\circ, 270^\circ]$.

A. NeRF Synthetic Dataset

a) *Setting.*: We evaluated iGaussian on 8 objects from the NeRF Synthetic Dataset, comparing it with INeRF [1] and iComMa [2]. INeRF parameters were set to `batch_size=2048` and `sampling_strategy=interest_regions`. The test datasets was generated by applying translation and rotation transformations to source poses, rendering target images using a pre-trained 3D Gaussian model trained for 30,000 iterations. For each scene, 300 target images were generated: 100 with angular errors in $\pm[20^\circ, 40^\circ]$ and 100 in $\pm[40^\circ, 80^\circ]$, and 100 in $\pm[80^\circ, 180^\circ]$ with a translation offset of $[-0.2, 0.2]$. INeRF and iComMa used target-source image pairs, while iGaussian required only the target image. INeRF and iComMa had a maximum of 300 optimization steps, while iGaussian used a Gaussian sampling strategy with $R = 4$ and camera viewpoints directed toward the scene origin.

TABLE I. Comparison of Accuracy and Runtime on the NeRF Synthetic Dataset. Our method maintains superior success rates and the fastest inference across angular ranges, demonstrating strong robustness in low-overlap views.

Methods	$\pm[20^\circ, 40^\circ]$			$\pm[40^\circ, 80^\circ]$			$\pm[80^\circ, 180^\circ]$		
	iNeRF	iComMa	Ours	iNeRF	iComMa	Ours	iNeRF	iComMa	Ours
Success Rate (%)	62.94	78.69	82.06	17.5	45.44	82.31	0	0	81.86
Time (s)	23.33	3.95	0.38	26.7	10.43	0.38	–	–	0.38

TABLE II. Pure Rotation Test on the Mip-NeRF360 Dataset. The error metrics for SIFT and SuperPoint are computed exclusively on successful correspondences. Their failure rates exceed 50%.

Methods	$\pm[20^\circ, 40^\circ]$			$\pm[40^\circ, 80^\circ]$			$\pm[80^\circ, 180^\circ]$		
	Avg.↓	Med.↓	$\leq 15^\circ \uparrow$	Avg.↓	Med.↓	$\leq 15^\circ \uparrow$	Avg.↓	Med.↓	$\leq 15^\circ \uparrow$
SIFT	18.9	3.13	22.4	38.8	13.8	5.7	–	–	0
SuperPoint	6.38	1.79	16.5	6.8	6.85	2.3	–	–	0
LoFTR	3.36	0.84	92.3	28.62	9.24	57.4	48.2	42.7	1.5
8-Point ViT	4.56	2.98	94.1	9.82	4.79	76.4	32.43	17.23	44.6
FAR	1.87	0.34	100	4.48	1.16	94.4	22.61	14.58	52.8
Ours	0.57	0.22	100	0.54	0.23	100	0.55	0.23	100

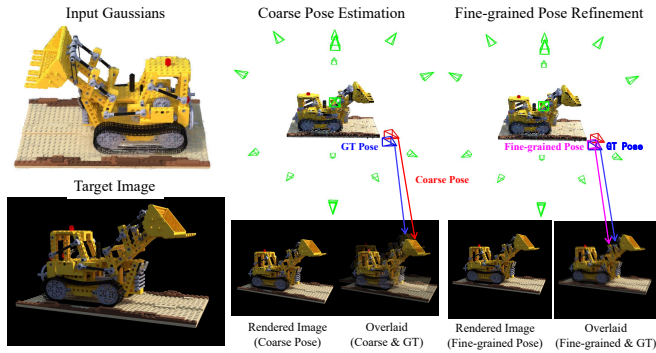


Fig. 4: Results of our two-stage pose estimation strategy. The first stage employs Gaussian Rendering-Based Coarse-Grained Pose Estimation to estimate coarse camera poses, while the second stage applies Correspondence-Based Pose Optimization for refinement. Blue camera poses denote ground truth, with red and purple coordinates representing coarse estimates and fine-grained optimizations, respectively. The comparison between rendered images (solid) and target images (ghosted) visualizes alignment accuracy.

b) Pose Estimation Results: Our method was evaluated in pose estimation tasks, measuring success rates with rotation errors $< 5^\circ$ and translation errors < 5 cm [42]. The experimental results are shown in Table I. In high-overlap views ($\pm[20^\circ, 40^\circ]$), it outperformed iNeRF and iComMa in success rate and reduced runtime by nearly tenfold compared to iComMa. In low-overlap ($\pm[40^\circ, 80^\circ]$) and extremely low-overlap ($\pm[80^\circ, 180^\circ]$) scenarios, our method’s advantages were more pronounced, maintaining stable success rates and runtime, while iNeRF and iComMa’s success rates dropped significantly due to their optimization-based strategies. Our method exhibited stronger robustness and efficiency. Specifically, the qualitative results are shown in Fig. 4.

B. Mip-NeRF 360 Dataset

a) Setting: We evaluated our method on nine scenes from the Mip-NeRF 360 dataset, generating 300 test images per scene (100 each for $\pm[20^\circ, 40^\circ]$ and $\pm[40^\circ, 80^\circ]$

$\pm[80^\circ, 180^\circ]$ angle deviations). Using a 15° rotation error threshold, we computed median and mean prediction errors to analyze performance. The Gaussian sampling strategy we adopted is set with $R = 0.1$, and our results outperform the existing SOTA.

b) Pose Estimation Results: The experimental results in Table II show that our method outperforms FAR [16] method in both high-overlap and low-overlap views for pure rotation estimation. In low-overlap cases, FAR struggles with increased errors due to sparse matching points, while our method regresses a precise initial pose and optimizes it using a render-and-compare strategy, significantly reducing errors. Additionally, LoFTR [22] excels in low-parallax views but lacks translation scale information, so we integrated 8-Point ViT [12] for scale estimation. Our method demonstrates robustness under large disparities and achieves lower pose estimation errors than existing approaches.

C. T&T+DB Dataset

a) Setting: We evaluated our method on four scenes from the T&T+DB dataset, generating the test dataset using the Mip-NeRF 360 strategy with Gaussian sampling ($R=0.1$) and viewpoints facing away from the origin. We investigated the performance of the model under the conditions of a 15° rotation error threshold and a 0.5m translation error threshold. We compared our method with solver-based (LoFTR, SuperGlue) and learning-based (ViT-8-Point), as well as plane mapping and optimization-based methods (NOPE-SAC [14]). Additionally, we benchmarked against FAR, which integrates optimization and learning paradigms and outperforms standalone methods, making it a key reference in our study.

b) Pose Estimation Results: Table III shows our method’s performance on the T&T+DB dataset. FAR, combining matching and solver-based paradigms, excels in various metrics, while end-to-end methods like 8-Point ViT perform well in translation estimation, and solver-based methods like LoFTR lead in rotation estimation. Our method surpasses FAR, reducing mean and median translation errors by $\sim 50\%$ (0.24 to 0.11 and 0.15 to 0.08) and rotation errors by $\sim 50\%$ (1.23 to 0.43) in the $\pm[20^\circ, 40^\circ]$ range. In large disparity views, it outperforms FAR by over $8\times$ in both translation and rotation, demonstrating high accuracy, robustness, and generalization under extreme viewpoint changes.

D. Ablation Study

a) Setting: We conducted an ablation study on the T&T+DB dataset to analyze key components of our method: Pose Attention, Weight Predictor Module(WMP), ViT [12],

TABLE III. Evaluation of Rotational-Translational Performance on the T&T+DB Dataset. End-to-end methods (ViT-8-Point) excel in translation estimation but exhibit deficiencies in rotation estimation. Solver-based methods (LoFTR) perform well in rotation estimation. Our hybrid approach strikes a balance between the two modalities and achieves improvements of over 30% in both translation and rotation performance compared to FAR, which also combines the two modalities.

Methods	$\pm[20^\circ, 40^\circ]$						$\pm[40^\circ, 80^\circ]$						$\pm[80^\circ, 180^\circ]$					
	Translation (m)			Rotation ($^\circ$)			Translation (m)			Rotation ($^\circ$)			Translation (m)			Rotation ($^\circ$)		
	Avg. ↓	Med. ↓	$\leq 0.5m$ ↑	Avg. ↓	Med. ↓	$\leq 15^\circ$ ↑	Avg. ↓	Med. ↓	$\leq 0.5m$ ↑	Avg. ↓	Med. ↓	$\leq 15^\circ$ ↑	Avg. ↓	Med. ↓	$\leq 0.5m$ ↑	Avg. ↓	Med. ↓	$\leq 15^\circ$ ↑
SuperGlue	—	—	—	3.88	0.59	97	—	—	—	17.4	9.33	68	—	—	0	—	—	0
NOPE-SAC-Reg	0.34	0.26	90.4	2.77	0.82	98.2	0.47	0.35	75.8	7.53	2.49	92.4	1.15	0.94	7.2	34.6	29.5	14.1
LoFTR+Reg.Scale	0.44	0.28	73.2	2.92	0.64	96.8	0.78	0.52	46.3	13.9	5.14	80.1	1.41	1.35	2.5	49.2	46.5	1.3
8-Point ViT	0.38	0.28	87.3	3.42	1.83	97.9	0.52	0.33	72.3	8.23	3.85	86.6	0.92	0.84	12.3	28.23	23.85	26.6
FAR	0.24	0.15	100	1.23	0.38	100	0.38	0.21	100	3.07	0.86	100	0.74	0.57	45.9	20.1	14.6	52.7
Ours	0.11	0.08	100	0.43	0.2	100	0.09	0.07	100	0.45	0.2	100	0.09	0.07	100	0.45	0.2	100

TABLE IV. Ablation results on different components of our approach.

Pose Attention	WMP	ViT	LoFTR + RANSAC	Translation (m)		Rotation ($^\circ$)	
				Avg. ↓	Med. ↓	Avg. ↓	Med. ↓
✓	×	×	×	0.38	0.3	5.18	4.94
✓	✓	×	×	0.29	0.23	3.95	3.63
✓	✓	✓	×	0.27	0.22	2.26	2.18
✓	✓	×	✓	0.19	0.13	0.43	0.2
✓	✓	✓	✓	0.11	0.08	0.43	0.2

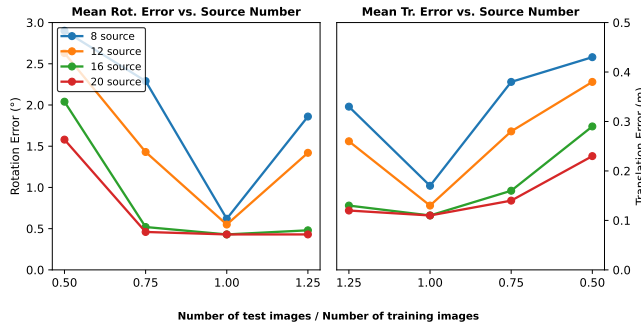


Fig. 5: Generalization Test on T&T+DB. We evaluate the model’s performance under varying numbers of input images to determine the optimal quantity for training and assess its robustness to input variations.

and Matching + Solver (LoFTR + RANSAC). For Stage 1, we evaluated initial pose regression using Gaussian scene priors. For ViT ablation, we assessed its role in refining translation scale estimation by applying the Stage 1 (Rendering-Based Coarse-Grained Pose Estimation) pose (T_{coarse}) as the scale factor. For Matching + Solver, we examined its impact on final pose accuracy. In the WMP ablation, we sampled only one source image to assess its effect on matching quality. All experiments were conducted on image pairs with $\pm[20^\circ, 40^\circ]$ angular deviation for consistency.

b) Ablation Study Results: Table IV demonstrates the impact of different modules on our method’s performance, ✓ indicates the module is used, and × indicates it is not. Key findings reveal that using only Pose Attention with a single source image for coarse pose regression limits accuracy due to insufficient viewpoints, while adding WMP and increasing spatial sampling to 16 source images reduces

translation and rotation errors by nearly 30%, highlighting the importance of broad spatial sampling. ViT significantly improves translation scale estimation by analyzing the relative pose between the observation image and the rendered image, providing accurate scale correction. Additionally, removing the feature matching and solver module reduces prediction accuracy by nearly 50%, emphasizing its critical role, particularly in low-disparity views.

c) Generalization Test: As show in Fig. 5. We evaluate our model’s generalization capability under varying numbers of source images. We tested sampling strategies with 8, 12, 16 and 20 source images, adjusting polar (ϕ) and azimuth (ψ) angles for each. Additionally, we tested views where input source images were 1/2, 3/4, and 3/2 of the original training setting. As the number of source images increases, the model performance improves. When there are 16 source images, the model achieves a balance between performance and resource usage. While performance fluctuates with varying input counts, the model remains robust with 16 or 20 images. However, insufficient source images lead to significant performance degradation, highlighting the importance of adequate viewpoint coverage for accurate pose estimation.

V. CONCLUSION

In this paper, we propose iGaussian, a novel real-time camera pose estimation method based on feed-forward 3D Gaussian inversion. Our approach integrates a multi-view pose regression network with Gaussian scene priors and a matching and solver module to seamlessly connect coarse pose estimation and fine optimization in a single feed-forward pass, bypassing the computationally expensive iterative render-compare process used in traditional methods. Experiments show that iGaussian performs excellently in rotation and translation prediction, and achieves a real-time speed of 2.87 FPS on mobile robots—nearly 10× faster than optimization-driven techniques. iGaussian balances accuracy and efficiency, providing a solution for visual navigation, robot localization, and AR.

REFERENCES

- [1] L. Yen-Chen, P. Florence, J. T. Barron, A. Rodriguez, P. Isola, and T.-Y. Lin, “in3r: Inverting neural radiance fields for pose estimation,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1323–1330, IEEE, 2021.

- [2] Y. Sun, X. Wang, Y. Zhang, J. Zhang, C. Jiang, Y. Guo, and F. Wang, "icomm: Inverting 3d gaussian splatting for camera pose estimation via comparing and matching," *arXiv preprint arXiv:2312.09031*, 2023.
- [3] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.
- [4] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023.
- [5] L. Liu, J. Gu, K. Zaw Lin, T.-S. Chua, and C. Theobalt, "Neural sparse voxel fields," *Advances in Neural Information Processing Systems*, vol. 33, pp. 15651–15663, 2020.
- [6] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, "pixelnerf: Neural radiance fields from one or few images," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4578–4587, 2021.
- [7] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocation," in *Proceedings of the IEEE international conference on computer vision*, pp. 2938–2946, 2015.
- [8] V. Balntas, S. Li, and V. Prisacariu, "Relocnet: Continuous metric learning relocation using neural nets," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 751–767, 2018.
- [9] S. Brahmbhatt, J. Gu, K. Kim, J. Hays, and J. Kautz, "Geometry-aware learning of maps for camera localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2616–2625, 2018.
- [10] Z. Teed and J. Deng, "Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras," *Advances in neural information processing systems*, vol. 34, pp. 16558–16569, 2021.
- [11] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.
- [12] C. Rockwell, J. Johnson, and D. F. Fouhey, "The 8-point algorithm as an inductive bias for relative pose prediction by vits," in *2022 International Conference on 3D Vision (3DV)*, pp. 1–11, IEEE, 2022.
- [13] S. En, A. Lechervy, and F. Jurie, "Rpnet: An end-to-end network for relative camera pose estimation," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pp. 0–0, 2018.
- [14] B. Tan, N. Xue, T. Wu, and G.-S. Xia, "Nope-sac: Neural one-plane ransac for sparse-view planar 3d reconstruction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 12, pp. 15233–15248, 2023.
- [15] S. Qian, L. Jin, and D. F. Fouhey, "Associative3d: Volumetric reconstruction from sparse views," in *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16*, pp. 140–157, Springer, 2020.
- [16] C. Rockwell, N. Kulkarni, L. Jin, J. J. Park, J. Johnson, and D. F. Fouhey, "Far: Flexible accurate and robust 6dof relative camera pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19854–19864, 2024.
- [17] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, pp. 91–110, 2004.
- [18] P.-E. Sarlin, A. Unagar, M. Larsson, H. Germain, C. Toft, V. Larsson, M. Pollefeys, V. Lepetit, L. Hammarstrand, F. Kahl, *et al.*, "Back to the feature: Learning robust camera localization from pixels to pose," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3247–3257, 2021.
- [19] H. C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections," *Nature*, vol. 293, no. 5828, pp. 133–135, 1981.
- [20] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 6, pp. 756–770, 2004.
- [21] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4938–4947, 2020.
- [22] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "Loftr: Detector-free local feature matching with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8922–8931, 2021.
- [23] Q. Wang, J. Zhang, K. Yang, K. Peng, and R. Stiefelhagen, "Matchformer: Interleaving attention in transformers for feature matching," in *Proceedings of the Asian Conference on Computer Vision*, pp. 2746–2762, 2022.
- [24] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox, "Deepim: Deep iterative matching for 6d pose estimation," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 683–698, 2018.
- [25] E. Kolker, R. Higdon, W. Haynes, D. Welch, W. Broomall, D. Lancet, L. Stanberry, and N. Kolker, "Moped: model organism protein expression database," *Nucleic acids research*, vol. 40, no. D1, pp. D1093–D1099, 2012.
- [26] W. Goodwin, S. Vaze, I. Havoutis, and I. Posner, "Zero-shot category-level object pose estimation," in *European Conference on Computer Vision*, pp. 516–532, Springer, 2022.
- [27] X. He, J. Sun, Y. Wang, D. Huang, H. Bao, and X. Zhou, "Onepose++: Keypoint-free one-shot object pose estimation without cad models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 35103–35115, 2022.
- [28] Z. Fan, P. Pan, P. Wang, Y. Jiang, D. Xu, H. Jiang, and Z. Wang, "Pope: 6-dof promptable pose estimation of any object," *Any Scene, with One Reference*, *arXiv*, vol. 2305, 2023.
- [29] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys, "Nice-slam: Neural implicit scalable encoding for slam," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12786–12796, 2022.
- [30] M. M. Johari, C. Carta, and F. Fleuret, "Eslam: Efficient dense slam system based on hybrid representation of signed distance fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17408–17419, 2023.
- [31] A. Rosinol, J. J. Leonard, and L. Carlone, "Nerf-slam: Real-time dense monocular slam with neural radiance fields," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3437–3444, IEEE, 2023.
- [32] H. Matsuki, R. Murai, P. H. Kelly, and A. J. Davison, "Gaussian splatting slam," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18039–18048, 2024.
- [33] S. Zhu, R. Qin, G. Wang, J. Liu, and H. Wang, "Semgauss-slam: Dense semantic gaussian splatting slam," *arXiv preprint arXiv:2403.07494*, 2024.
- [34] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield, "Deep object pose estimation for semantic robotic grasping of household objects," *arXiv preprint arXiv:1809.10790*, 2018.
- [35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
- [36] J. D. Foley, *Computer graphics: principles and practice*, vol. 12110. Addison-Wesley Professional, 1996.
- [37] H. Ren, Y. Yang, H. Wang, B. Shen, Q. Fan, Y. Zheng, C. K. Liu, and L. J. Guibas, "Adela: Automatic dense labeling with attention for viewpoint shift in semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8079–8089, 2022.
- [38] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [39] P. H. Torr and A. Zisserman, "Mlesac: A new robust estimator with application to estimating image geometry," *Computer vision and image understanding*, vol. 78, no. 1, pp. 138–156, 2000.
- [40] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Mip-nerf 360: Unbounded anti-aliased neural radiance fields," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5470–5479, 2022.
- [41] P. Hedman, J. Philip, T. Price, J.-M. Frahm, G. Drettakis, and G. Brostow, "Deep blending for free-viewpoint image-based rendering," *ACM Transactions on Graphics (ToG)*, vol. 37, no. 6, pp. 1–15, 2018.
- [42] T. Hodan, F. Michel, E. Brachmann, W. Kehl, A. GlentBuch, D. Kraft, B. Drost, J. Vidal, S. Ihke, X. Zabulis, *et al.*, "Bop: Benchmark for 6d object pose estimation," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 19–34, 2018.