Enhancing Visual Relocalization with Dense Scene Coordinates Derived from 3D Gaussian Splatting

Jou-Min Liu[1], Hsuan-Kung Yang[2], Tsung-Chih Chiang[1], Ting-Ru Liu[1], Chun-Wei Huang[1], Quan Kong[2], Norimasa Kobori[2], and Chun-Yi Lee[1]

***Abstract*— Scene coordinate regression is a visual localization method that directly regresses the 3D scene coordinate for a set of pixels. Although existing works have demonstrated the feasibility to learn 3D scene coordinates from RGB images with ground truth poses, their effectiveness is limited by the availability of training data, particularly due to the absence of 3D information. To address this limitation, we introduce a novel three-stage approach for SCR model learning from 2D data. Our method begins by employing 3D Gaussian Splatting for generating a dense reconstruction of the scene. Subsequently, the SCR model is initialized with pseudo scene coordinates derived from the reconstruction. Finally, the model is refined using a sparse set of real images to mitigate the domain gap between pseudo scene coordinates and real scene coordinates. Our approach is validated through comprehensive experiments, resulting in performance improvements on the DL3DV-10K and 7 Scenes datasets.**

I. INTRODUCTION

RGB-based scene coordinate regression (SCR) for camera pose estimation aims to learn the coordinates of a scene from a collection of 2D camera views within a 3D space [1], [2]. These learned coordinates are then employed to calculate the camera pose when presented with a novel camera view. In real-world scenarios, as the dense ground truth of 3D coordinates is not easy to obtain, previous studies have resorted to the strategy of projecting the estimated 3D coordinates back onto the 2D image plane and evaluating the reprojection error [1], [2]. Nonetheless, this reliance on reprojection errors as the sole optimization criterion may result in suboptimal SCR performance since the 3D coordinates are implicitly learned from multi-view images. While certain investigations [3] have endeavored to refine sparse scene coordinates obtained through structure-from-motion (SfM) techniques [4], [5], they may be inadequate for methodologies necessitating densely populated scene coordinates for precise camera pose prediction. As a result, the development of a strategy to enhance the availability of dense scene coordinate data emerges as a pivotal and indispensable requirement.

Recent advancements in 3D Gaussian splatting (3DGS) [6] present a compelling solution to address the challenges mentioned above. This technique can encode and expand sparse point clouds into informative splats, and enables dense reconstruction of the scene with high fidelity. More specifically, these derived splats are explicitly encoded in the 3D space, and therefore offer more comprehensive and detailed 3D coordinate information. Compared to the sparse point clouds from SfM, the use of 3DGS in SCR unlocks new possibilities for utilizing the data they contain to improve and refine scene coordinate estimation and provides a promising opportunity to enhance the training process of SCR.

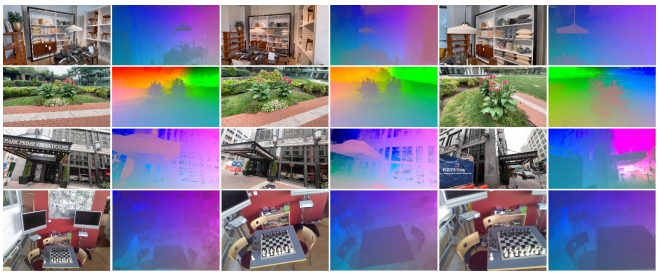

Fig. 1. An illustration of the images from the DL3DV-10K and 7 Scenes datasets, along with the pseudo scene coordinates derived from the constructed 3DGS model.

In light of the above potential, in this work, we develop a new training framework for Scene Coordinate Regression (SCR) that leverages the properties of 3D Gaussian Splatting (3DGS) and introduce the concept of pseudo scene coordinates (pSCs), which are explicit 3D coordinates inferred from the constructed 3DGS model, as depicted in Fig. 1. The framework is structured into three stages: (1) the 3DGS learning stage for deriving pSCs, (2) the scene coordinate initialization stage, and (3) the scene coordinate fine-tuning stage. Specifically, in the first stage, the 2D camera views are utilized to train a 3DGS model. Based on this model, the 3D coordinates of the derived Gaussian splats, which are dense in the 3D space, are utilized to derive pSCs. In the second stage, these pSCs generated by the 3DGS model are leveraged to train the SCR model at a scene-specific level during the scene coordinate initialization stage. In the scene coordinate fine-tuning stage, the trained model from the second stage is further fine-tuned with the real training sequence using the ground truth 2D camera poses. The design of these three stages enhances the SCR model's generalizability and enables the estimation of high-quality scene coordinates with greater accuracy. To validate the proposed framework, we evaluate our methodology on both outdoor and indoor datasets, including DL3DV-10K [7] and 7 Scenes [8], for evaluating the performance of our SCR approach. The experimental results validate that the proposed methodology can indeed provide benefits, and leads to more effective learning and better data efficiency when only lim-

[1] Elsa Lab, Department of Computer Science, National Tsing Hua University, Hsinchu City, Taiwan.
[2] Woven by Toyota, Inc., Japan.

ited 2D camera views are available. It also delivers promising performance in estimated translational and rotational errors.

II. PRELIMINARY

A. Scene Coordinate Regression (SCR)

SCR is a visual localization method that establishes correspondences between 2D pixels on an image and 3D coordinates in a scene by directly regressing the 3D scene coordinate for each pixel. The primary learning objective is to learn a mapping function $f(\cdot)$ such that $\mathcal{Y} = f(I)$, where $\mathcal{Y}$ represents the 3D scene coordinates, and I is the input image. After obtaining $\mathcal{Y}$, the camera pose can be estimated using the Perspective-n-Point (PnP) algorithm, expressed as:

$$\mathrm{h} = g^{PnP}(\mathcal{C}_{\mathcal{I}}), \text{with } \mathcal{C} = \{(\mathrm{p}_i, \mathrm{y}_i) | \mathrm{p}_i \in I, \mathrm{y}_i \in \mathcal{Y}\}, \quad (1)$$

where h denotes the estimated camera pose, $g^{PnP}(\cdot)$ represents the PnP algorithm augmented with RANSAC, $\mathcal{C}$ is the set of all 2D-3D correspondences, and $\mathcal{C}_{\mathcal{I}}$ denotes the subset of inlier correspondences. Inlier correspondences are those that are consistent with the estimated camera pose, in contrast to outliers, which fail to align accurately with the estimated pose. Recovering dense 3D information from 2D images is a highly challenging task. In early works [9], [10], the depth information was necessary to recover ground truth scene coordinates for training. DSAC++ [1] first demonstrated the feasibility of learning 3D scene coordinates solely from a series of RGB images annotated with ground truth poses for a given scene. This was achieved through implicit triangulation, constrained by multi-view geometry, and a carefully designed initialization process. Subsequent research endeavors [2], [11] simplified the procedure and improved both performance and robustness. Nevertheless, these methods still require a substantial number of camera views to effectively reconstruct the 3D information implicitly.

B. 3D Gaussian Splatting (3DGS)

3DGS is a real-time rendering method which explicitly models a 3D scene as a collection of anisotropic 3D Gaussians. In this context, a Gaussian splat (or simply a splat) represents a volumetric space data point characterized by a Gaussian function. Each Gaussian $G(x)$ encapsulates 3D information such as 3D coordinate and orientation, scale, opacity, and color information. A splat is defined by a mean $\mu \in \mathbb{R}^3$ and a covariance matrix Σ, formulated as follows:

$$G(x) = \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right). \quad (2)$$

The initial Gaussians are formed by SfM points, and the properties of these Gaussians are updated through gradients. They may be cloned or split at specified intervals if the accumulated gradients exceed a threshold. During the optimization process, the Gaussians are rendered onto a 2D image plane, where a tile-based rasterizer is employed to enhance rendering efficiency, expressed as follows:

$$C(v) = \sum_{i \in N} c_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j), \quad (3)$$

where c_i is the color attributed to the i-th Gaussian splat, N denotes the set of Gaussian splats within a tile, and $C(v)$ represents the rendered color at pixel v of the 2D camera plane. The term α_i is derived from the product $o_i \cdot G^i_{2D}(v)$, where o_i indicates the opacity of the i-th Gaussian splat, and $G^i_{2D}(\cdot)$ is the function which describes the projection of that Gaussian splat onto the 2D camera plane.

III. METHODOLOGY

A. Overview of the Framework

Fig. 2 illustrates an overview of the proposed SCR framework, which adopts a comprehensive approach by integrating 3DGS to enhance visual re-localization. This framework encompasses three stages, as described in Section I. Beginning with a collection of multi-view 2D RGB images, the framework employs SfM to generate a set of SfM points, which serve as the initial elements for training the 3DGS model, as described in Section II-B. The trained 3DGS model aims to use Gaussian splats as 3D pSCs for initiating the SCR model's training process. These images are processed by a scene-agnostic feature extractor [11], which works in tandem with a scene-specific regression head. The feature extractor is responsible for interpreting the general attributes of the scene, while the regression head customizes this interpretation to the specifics of the scene under consideration and generates scene coordinates. A key breakthrough of this framework is its endeavor to extend beyond the limitations of the originally provided 2D camera view data to densify the original set of 3D scene coordinates extracted by SfM. This process involves using the data to infer 3D insights through 3DGS and then leverage the extensive collection of Gaussian splats that bear rich 3D geometric details to enhance the SCR learning. By adopting 3DGS and SCR models together, our framework enhances the use of limited 2D camera view data more effectively and efficiently than methodologies that rely solely on the original 2D images. Please note that further discussion and evaluation on our framework's data efficiency are provided in Section IV.

B. Pseudo Scene Coordinate (pSC) Generation

Pseudo scene coordinates (pSCs) generated from 3DGS present an innovative direction for initializing the training stage of an SCR model. The adoption of these pseudo coordinates aims to exploit their rich, spatially-dense information to bridge the gap between the sparsity of 2D camera view data and the continuous nature of physical environments. The rationale behind employing pSCs from 3DGS is based on their capacity to capture the spatial relationships present within a scene. Such spatial details, including the depth and density information provided by 3DGS, can equip an SCR model with a more profound comprehension of the scene's 3D geometry. This aspect becomes essential especially in scenarios where depth information is unavailable or 2D camera views are sparse.

To derive pSCs for a specific pixel v on the camera plane, a mapping is established to identify all splat indices that contribute to rendering $C(v)$. This process considers the

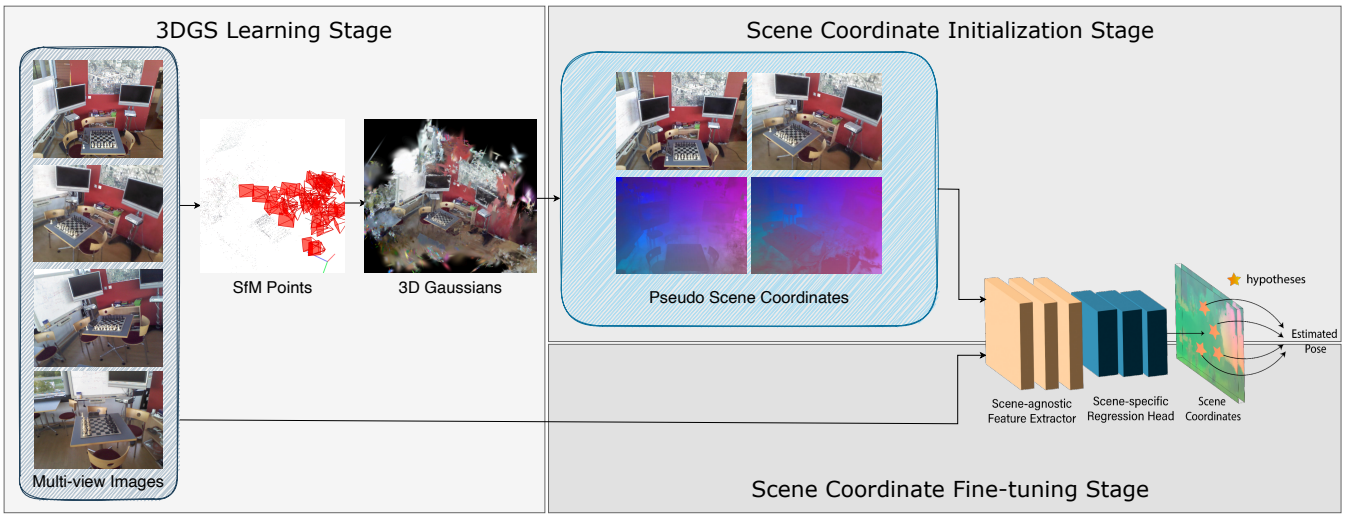

Fig. 2. An overview of the proposed framework, which encompasses three main stages.

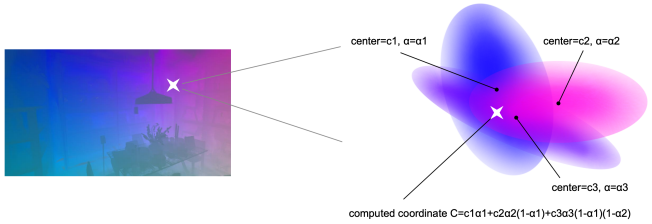

Fig. 3. A image of deriving the pSC from the means of a set of Gaussians.

influence of each Gaussian splat by taking into account their respective opacity values α_i. The pSC for v is then determined as a weighted sum of each Gaussian splat's contributions, which can be formulated as follows:

$$pSC(v) = \sum_{i \in N} \mu_i \cdot \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j), \tag{4}$$

where N denotes all splats contribute to render $C(v)$, $pSC(v)$ represents the weighted pseudo scene coordinate for pixel v, and μ_i denotes the mean 3D coordinate of the i-th splat, as depicted in Fig. 3. This approach is pivotal for synthesizing scene coordinates not explicitly present in the original 2D dataset, which enhances the 3D coordinate space information for SCR model training.

C. Scene Coordinate Regression Training Schedule

In order to leverage the pSCs, we design a two-stage training procedure for our SCR methodology, which includes: (a) a scene coordinate initialization stage (denoted as SC_{init}) and (b) a scene coordinate fine-tuning stage (denoted as SC_{finetune}). During the SC_{init} stage, the SCR model is trained with the aid of pSCs. This is intended to enhance the model's capacity for comprehensive scene representation. At the SC_{finetune} stage, the trained model from the SC_{init} stage is further fine-tuned with the real training sequence using the ground truth camera poses. The SC_{init} stage is beneficial, since it allows the model to efficiently learn scene-specific features in detail. However, the SC_{finetune} stage remains essential to address the domain gap between pseudo scene coordinates and real scene coordinates. More specifically, in this work, we train the SCR model firstly with SC_{init} followed by SC_{finetune}, with 16 epochs for each stage. For the SC_{init} stage, we optimize a composite loss function that includes: (a) a L2 loss derived from the l2-norm of the difference between camera coordinates converted from the estimated scene coordinates and the pSCs, and (b) a robust reprojection loss L_{reproj} introduced in ACE [11]. The training loss employed for SC_{init} can be formulated as follows:

$$L_{\text{init}} = \sum_{i \in I} \ell_{\text{reproj}}(\mathrm{p}_i, \mathrm{y}_i, \mathrm{h}^*) + \left\| \mathrm{h}^{*-1} pSC(\mathrm{p}_i) - \mathrm{h}^{*-1} \mathrm{y}_i \right\|_2^2 \tag{5}$$

$$\ell_{\text{reproj}}(\mathrm{p}_i, \mathrm{y}_i, \mathrm{h}^*) = \begin{cases} \left\| \mathrm{p}_i - K \mathrm{h}^{*-1} \mathrm{y}_i \right\| & \text{if } \mathrm{y}_i \in \mathcal{V} \\ \left\| \mathrm{y}_i - \hat{\mathrm{y}}_i \right\| & \text{otherwise.} \end{cases}, \tag{6}$$

where K is the camera intrinsic, h^* is the ground truth camera pose, $\mathcal{V}$ is the set of valid scene coordinate predictions that are between 10cm and 1,000m in front of the image plane and have a reprojection error below 1,000 pixels, and for invalid predictions, a dummy scene coordinate $\hat{\mathrm{y}}_i$ is used, which is 10m in front of the camera calculated from the ground truth camera pose. For the SC_{finetune} stage, the model is fine-tuned with only the reprojection loss, formulated as:

$$L_{\text{finetune}} = \sum_{i \in I} \ell_{\text{reproj}}(\mathrm{p}_i, \mathrm{y}_i, \mathrm{h}^*). \tag{7}$$

IV. EXPERIMENTAL RESULTS

A. Experimental Setups

We evaluate our methodology on both outdoor and indoor datasets, including the DL3DV-10K [7] and 7 Scenes [8] datasets. From DL3DV-10K, which offers a continuous sequence for each scene, we selected 11 sample scenes. To examine our method, we conduct experiments with a limited amount of training data. For DL3DV-10K, we sample the

TABLE I

COMPARISON OF PERFORMANCE FOR THE SC_{INIT} STAGE USING DIFFERENT METHODS FOR GENERATING PSCS.

Pseudo Scene Coordinate	DL3DV-10K			7 Scenes		
	(deg)	(cm)	(%)	(deg)	(cm)	(%)
None (ACE) [11]	0.15	14.29	36.55%	1.13	3.66	72.39%
Sparse (SfM)	0.15	14.00	36.81%	1.15	3.63	72.46%
Dense (3DGS)	0.13	11.87	44.87%	1.11	3.41	74.71%

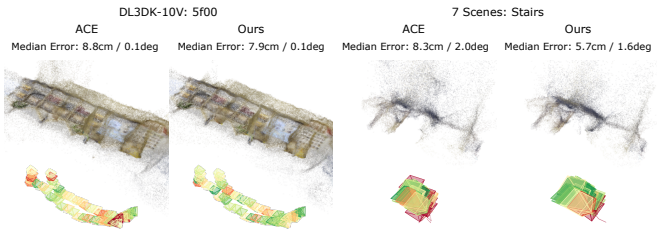

Fig. 4. The comparison of predicted scene coordinates and camera poses on DL3DV-10K. From the clear point clouds, it is evident that our method learns scene coordinates with greater accuracy, resulting in superior performance on camera pose estimation.

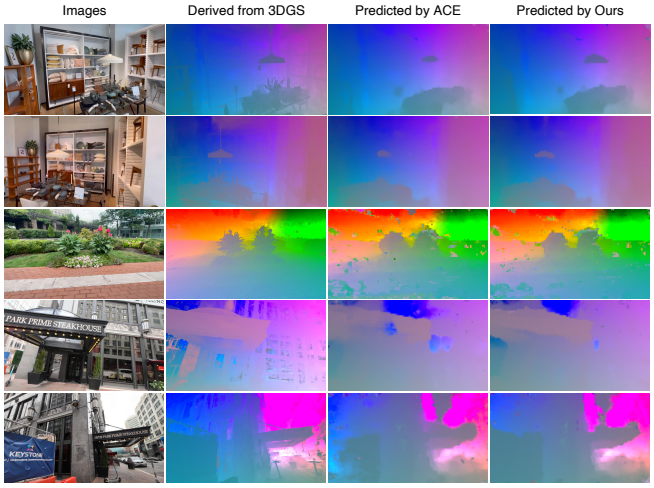

Fig. 5. An illustration of the images from the DL3DV-10K dataset, along with the pSCs derived from the constructed 3DGS model and the scene coordinates predicted by ACE and our methodology.

training data every ten frames, and construct a dataset with only 10% of the original training set size, while the rest data are used for testing. As for 7 Scenes, we sample the training data every fifty frames from the training sequences, while the test data remain unchanged.

B. Effectiveness of Pseudo Scene Coordinates

In this section, we aim to validate the efficacy of the pre-training stage utilizing pseudo scene coordinates (pSCs) for the SCR model. This experiment focuses on the median errors and accuracy of the pSCs generated by various methods during the initial training phase. We begin by utilizing a subset of the training data (10% of the DL3DV dataset and 2% of the 7scene dataset) to construct sparse point clouds using COLMAP [4], [5], an SfM approach. These sparse point clouds, derived from limited camera views, serve as the foundation for training our 3DGS model, which is used to compute pSCs for training our SCR model. Table I presents the evaluation results. In all experimental setups, the SCR model undergoes the SC_{finetune} stage that is without the inclusion of the camera coordinate loss derived from pSCs. Three configurations are evaluated: (a) not using pSCs (i.e., the original ACE baseline), (b) employing sparse pSCs obtained directly derived from point clouds constructed with COLMAP, and (c) generating dense pSCs using 3DGS. The results of this analysis are presented in Table I. It demonstrates the precision of pseudo scene coordinate generation and the impact of using dense versus sparse scene coordinates. This assessment highlights the potential of integrating high-quality and dense pSCs to enhance model performance in the SCR task.

C. Qualitative Results

1) Visualization of Camera Pose: Fig. 4 illustrate predicted scene coordinates with point clouds and camera poses on DL3DV-10K datasets. It is clear that point clouds of our method have less sparse areas, this implies that our method provide a more consistent scene coordinate, sequentially resulting in more precise camera pose estimation.

2) Comparison of Predicted Scene Coordinates: To evaluate the effectiveness of incorporating pSCs in training the SCR model, we visualize the scene coordinates predicted by the SCR models. As depicted in Fig. 5, it can be observed that our model predicts scene coordinates with clearer boundaries. The predicted scene coordinates for the same objects exhibit a higher degree of similarity, and the predictions contain reduced noise compared to the baseline.

V. CONCLUSIONS

We proposes a novel three-stage approach to address the challenge of limited training data for scene coordinate regression (SCR) models used in visual localization. By leveraging 3D Gaussian splatting to generate a dense scene reconstruction from 2D data, the method can derive pseudo scene coordinates to initialize the SCR model. The model is then refined using a sparse set of real images to bridge the domain gap between pseudo scene coordinates and real scene coordinates. Comprehensive experiments on the DL3DV-10K and 7 Scenes datasets demonstrate that this approach can effectively learn a SCR model which rely solely on limited real training data. The proposed technique provides an effective way to train accurate SCR models for visual localization tasks even when only 2D data is available, overcoming a key limitation of previous methods. Overall, this work highlights the potential of combining learning-based techniques with 3D reconstruction to tackle challenging problems in visual localization with limited training images.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support from the National Science and Technology Council (NSTC) in Taiwan under grant numbers MOST 111-2223-E-007-004-MY3, as well as the financial support from Woven by Toyota,

Inc., Japan. The authors would also like to express their appreciation for the donation of the GPUs from NVIDIA Corporation and NVIDIA AI Technology Center (NVAITC) used in this work. Furthermore, the authors extend their gratitude to the National Center for High-Performance Computing (NCHC) for providing the necessary computational and storage resources.

REFERENCES

[1] E. Brachmann and C. Rother, "Learning less is more - 6D camera localization via 3D surface regression," in *CVPR*, 2018.

[2] ——, "Visual camera re-localization from RGB and RGB-D images using DSAC," *TPAMI*, 2021.

[3] S. T. Nguyen, A. Fontan, M. Milford, and T. Fischer, "Focustune: Tuning visual localization through focus-guided sampling," in *WACV*, January 2024, pp. 3606–3615.

[4] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[5] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *European Conference on Computer Vision (ECCV)*, 2016.

[6] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, no. 4, July 2023. [Online]. Available: https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/

[7] L. Ling, Y. Sheng, Z. Tu, W. Zhao, C. Xin, K. Wan, L. Yu, Q. Guo, Z. Yu, Y. Lu *et al.*, "Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision," *arXiv preprint arXiv:2312.16256*, 2023.

[8] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, "Scene coordinate regression forests for camera relocalization in rgb-d images," in *CVPR*, 2013.

[9] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. W. Fitzgibbon, "Scene coordinate regression forests for camera relocalization in rgb-d images," *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2930–2937, 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:8632684

[10] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother, "Dsac — differentiable ransac for camera localization," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2492–2500, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:4001530

[11] E. Brachmann, T. Cavallari, and V. A. Prisacariu, "Accelerated coordinate encoding: Learning to relocalize in minutes using rgb and poses," in *CVPR*, 2023.