Unlocking Single-View Constraints for Efficient Camera Relocalization with Keypoint-Level Multi-View Geometric Consistency in Training

Supplementary Material

1. Introduction

As discussed in the main paper, camera relocalization aims to recover the 6-DoF pose of a query image within a known 3D scene. While strong multi-view geometric constraints can significantly enhance accuracy, directly enforcing image-level multi-view consistency is often computationally prohibitive. Our approach circumvents this limitation by shifting from dense image supervision to sparse, geometry-aware keypoint correspondences, and by enriching the available viewpoints through 3D Gaussian splatting. In this supplementary material, we provide additional technical details and extended results that support the main paper.

2. More Details

2.1. Training Strategy with Multi-View Constraints

To achieve sufficient training efficiency, our training strategy employs multi-view constraints only at the initial and final phases of training, thereby significantly reducing computational overhead. We additionally experimented with imposing the multi-view loss over the entire training process; however, this increased the training time to more than 3 hours while providing only a 0.15% performance gain under the (5 deg, 5cm) threshold, which we consider negligible in light of the substantial additional computational cost.

2.2. Algorithm Pseudocode

In Algorithm 1, Algorithm 2, Algorithm 3, and Algorithm 4 we present the complete training pipeline. First, we extract 3D scene keypoints by evaluating SfM-based match scores [12]. Next, we generate dense projections of these keypoints across all views and store indexed tuples for gradient decoupling. During each training iteration, we apply 3D Gaussian-splatting–based augmentation [10] to enrich viewpoint diversity, predict scene coordinates via a CNN, and then minimize a multi-view optimized bundle loss.

3. More Results

3.1. Extended Quantitative Results

We further introduce two additional ablation settings to better isolate and validate the contribution of our design choices: (1) training the ACE head on all SfM points; and (2) training the ACE head on the keypoints selected by the matching-rate criterion. The detailed results are reported in Table 1.

Algorithm 1 Stage 1: Acquisition of Scene Keypoints

Require: Sparse 3D points $\mathcal{X}$; multi-view observations $\mathcal{O}(X) = \{(\mathbf{x}_i, \mathbf{d}_i)\}_{i=1}^{n}$; per-view weights $\{w_i\}$; descriptor bandwidth σ; threshold τ
Ensure: High-confidence keypoint set $\mathcal{K}$

$\mathcal{K} \leftarrow \emptyset$
for all $X \in \mathcal{X}$ **do**
  // Compute mean descriptor as in Eq.2
  $\boldsymbol{\phi}(X) \leftarrow \frac{1}{n}\sum_{i=1}^{n} \mathbf{d}_i$
  // Compute matching score as in Eq.1
  $M(X) \leftarrow \sum_{i=1}^{n} w_i \exp\big(-\|\boldsymbol{\phi}(X) - \mathbf{d}_i\|_2^2/(2\sigma^2)\big)$
  if $M(X) \geq \tau$ **then**
    $\mathcal{K} \leftarrow \mathcal{K} \cup \{X\}$ // retain high-confidence keypoint
  end if
end for

Algorithm 2 Stage 2: Keypoint-Driven Multi-View Correspondences

Require: Keypoints $\mathcal{K}$; camera intrinsics K; camera poses $\{(R_i, t_i)\}$
Ensure: Dense multi-view projections $\{\mathbf{u}_i(X)\}$

for all $X \in \mathcal{K}$ **do**
  for all view i **do**
    // Projection as in Eq.4
    $\mathbf{u}_i(X) \leftarrow \pi(X; R_i, t_i) = K\,(R_i X + t_i)$
  end for
end for

Algorithm 3 Stage 3: Training Buffer Generation for Gradient Decoupling

Require: Keypoints $\mathcal{K}$; observed projections $\mathbf{u}_i^{\text{obs}}(X_k)$; ACE descriptors $\{\mathbf{d}_k\}$; camera poses $\{(R_i, t_i)\}$
Ensure: Training buffer $\mathcal{B}$

$\mathcal{B} \leftarrow \emptyset$
for all $k = 1, \ldots, |\mathcal{K}|$ **do**
  for all view i observing X_k **do**
    // Store keypoint-level tuple as in gradient decoupling paragraph
    Store $(k,\ \mathbf{d}_k,\ R_i, t_i,\ \mathbf{u}_i^{\text{obs}}(X_k))$ in $\mathcal{B}$
  end for
end for

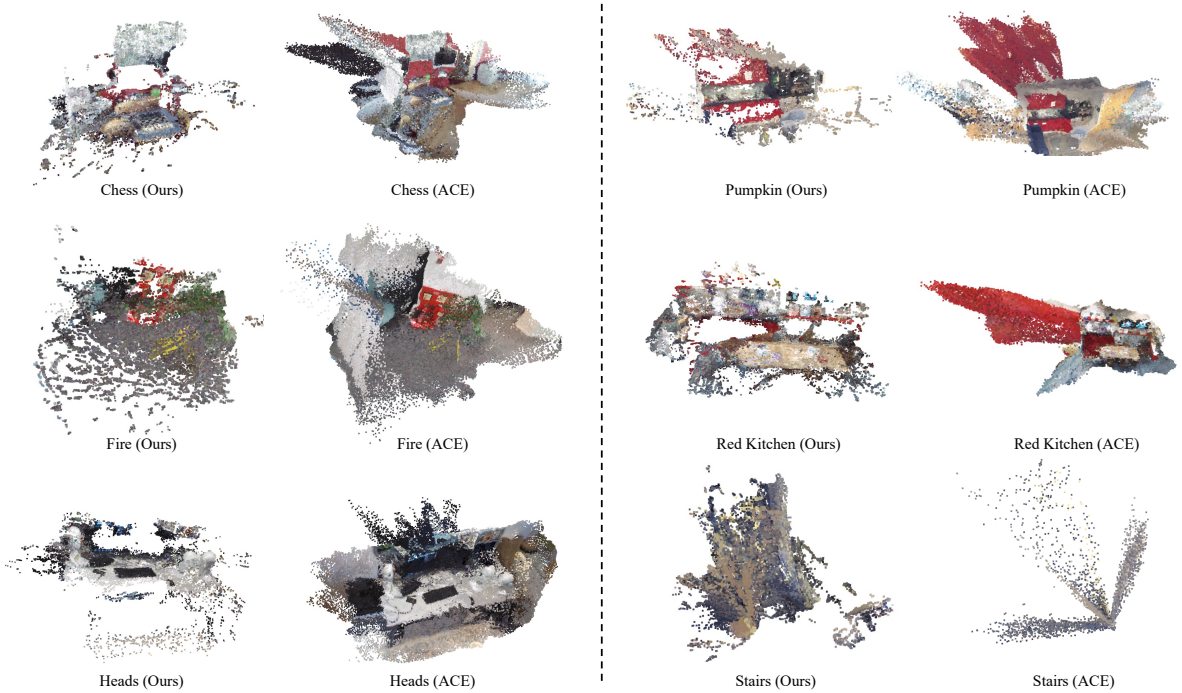

Figure 1. **Comparison of Scene Reconstruction on 7Scenes.** Since our method directly regresses scene coordinates, we perform scene recovery on the 7Scenes dataset after training to visualize the spatial structure learned by our network.

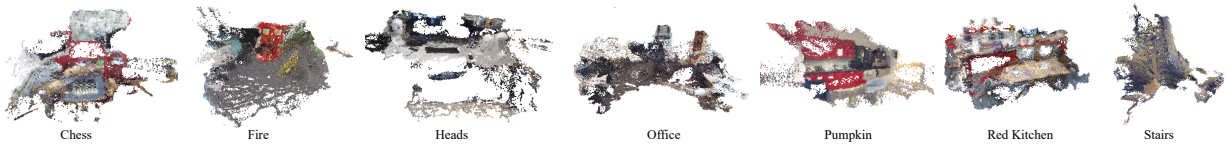

Figure 2. **Dense Scene Reconstruction on 7Scenes.**

In Table 3, we report the median translation and rotation errors of the poses estimated by our method with other methods: SCoRe Forest [13], DSAC [5], DSAC++ [3], HSC-Net [11], DSAC* [4], SANet [15], KFNet [16], DSM (single and video) [14], ACE [6], STDLoc [9]. By leveraging stronger observations, our approach further improves localization performance.

3.2. Test on Wayspots

In Figure 3, we visualize the cumulative distribution of pose errors, where each error is defined as the maximum of the rotation and translation errors.

We also evaluate our approach on the Wayspots [6] dataset, see Table 2. Wayspots is a publicly available relocalization benchmark constructed from the Map-free[1] mobile capture dataset, and consists of 10 scenes. Each scene depicts a small outdoor area and provides two complete, independent scans, with ground-truth camera poses obtained via SfM-based reconstruction.

Our method achieves the best performance on several scenes, including Bears, Tentrils, The Rock, and Map, and attains overall accuracy comparable to the ACE baseline. We further observe that the Wayspots scenes themselves exhibit limited viewpoint coverage and relatively sparse revisits of the same locations; in other words, many test viewpoints are poorly observed in the training scans. This lack of diverse observations can cause our method to fail in certain cases. In principle, incorporating the dual-branch EMR design could alleviate this issue by better handling such under-observed viewpoints.

3.3. Scene Reconstruction.

We visualize the learned scene coordinates after training to demonstrate that our method indeed learns more accurate scene coordinates. On the 7Scenes dataset, we compare the reconstructed scenes in Fig. 1. As can be observed, applying multi-view constraints during training effectively suppresses the learning of incorrect scene coordinates. Our

Table 1. Additional ablation study for independently validating our design choices. **SfM Points** and **MR Points** denote the number of training points under the all-SfM-point setting and the matching-rate-selected keypoint setting, respectively. **SfM + ACE Head** and **MR + ACE Head** report the corresponding localization performance when training the ACE head under the two settings.

Method	chess	fire	heads	office	pump.	redk.	stairs	Avg.
SfM Points	189,622	295,915	70,211	249,070	149,756	472,378	131,766	222,674
SfM + ACE Head	99.95%	87.45%	97.80%	95.65%	89.30%	89.86%	64.70%	89.24%
MR Points	6,791	4,644	4,679	5,200	3,523	10,713	4,165	5,674
MR + ACE Head	100.00%	99.95%	97.90%	99.48%	96.00%	99.02%	89.80%	97.45%
ACE	100.00%	99.50%	99.70%	100.00%	99.90%	98.20%	81.90%	97.03%
Ours	100.00%	100.00%	100.00%	99.60%	100.00%	99.90%	96.70%	99.45%

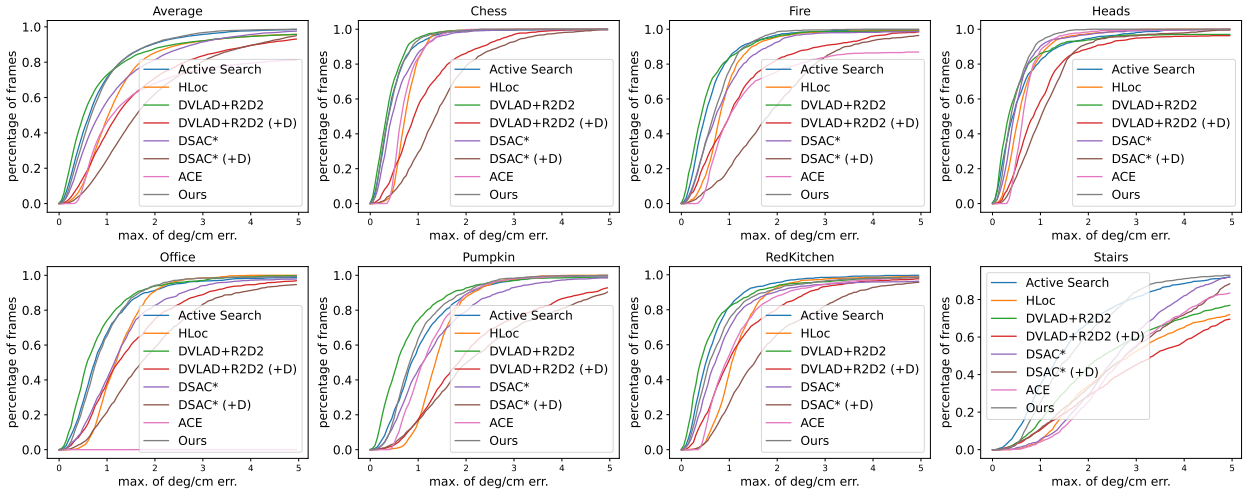

Figure 3. Cumulative distribution of pose errors, where each error is defined as the maximum of the rotation and translation errors in 7scenes.

Table 2. Performance evaluation on the Wayspots dataset (outdoor) with in (5 deg, 5 cm)

Scene	DSAC* (Tiny)	ACE	Ours
Cubes	68.70%	**97.00%**	86.96%
Bears	73.10%	80.70%	**86.21%**
Inscription	41.30%	**49.00%**	47.75%
The Rock	99.80%	**100%**	**100%**
Tendrils	19.60%	34.90%	**35.28%**
Map	53.30%	56.50%	**57.06%**
Square Bench	60.30%	**66.70%**	57.54%
Statue	0.00%	0.00%	0.00%
Lawn	20.00%	**35.80%**	34.54%
Average	48.46%	**57.84%**	56.15%

method eliminates most outliers. These outliers correspond to the failure mode described in the main paper, where the network predicts an arbitrary point along the viewing ray. If the network is not implicitly constrained on such predictions, these points remain as severe depth outliers.

On 7Scenes, we also find that our approach generalizes to image regions that are not used as keypoints. Even without keypoint guidance, our method produces noticeably denser scene reconstructions. We show these dense reconstructions in Fig. 2. Even under such densification, our method still constrains the reconstructed scene points to plausible locations, instead of producing exaggerated outliers.

We further conduct reconstruction comparisons on the Wayspots dataset, as shown in Fig. 4. Compared with indoor datasets, this dataset is substantially more challenging due to higher scene complexity and fewer distinctive visual features. Thanks to the multi-view constraints, our method can still effectively prevent scene points from degenerating into outliers, even in such difficult outdoor scenarios. The reconstructed scenes from our method almost always preserve recognizable objects and scene structures, whereas the baseline ACE method fails to produce meaningful scene reconstructions in most cases.

Table 3. Performance evaluation on the 7Scenes dataset (indoor), median error in cm/°, with scene coordinates regression methods.

Scene	Chess	Fire	Heads	Office	Pumpkin	Kitchen	Stairs	Average
SCoRe Forest	3.0/0.7	5.0/1.5	6.0/5.5	4.0/0.8	4.0/0.7	4.0/0.8	32.0/1.3	8.3/1.6
DSAC	2.0/0.7	3.0/1.0	2.0/1.3	3.0/1.0	5.0/1.3	5.0/1.5	190.0/49.4	30.0/8.0
DSAC++	2.0/0.5	2.0/0.9	1.0/0.8	3.0/0.7	4.0/1.1	4.0/1.1	9.0/2.6	3.6/1.1
HSC-Net	2.1/0.7	2.2/0.9	1.2/0.9	2.7/0.8	4.0/1.0	4.0/1.8	3.1/0.8	2.8/1.0
DSAC*	1.8/0.6	1.9/0.9	1.2/0.7	2.6/0.8	4.1/1.1	3.9/1.3	4.3/1.3	2.8/1.0
SANet	3.0/0.9	3.0/1.1	2.0/1.5	3.0/1.0	5.0/1.3	4.0/1.4	16.0/4.6	5.1/1.7
KFNet	1.8/0.7	2.3/0.9	1.4/0.8	2.5/0.7	3.7/1.0	3.8/1.2	3.3/0.9	2.7/0.9
DSM (sgl.)	2.0/0.7	2.0/0.9	1.0/0.9	3.0/0.8	4.0/1.2	4.0/1.2	5.0/1.3	3.0/1.0
DSM (vdo)	2.0/0.7	2.0/0.8	1.0/0.8	3.0/0.8	4.0/1.1	3.0/1.1	4.0/1.2	2.7/0.9
ACE	0.6/0.2	0.8/0.3	0.5/0.3	1.1/0.3	1.1/0.2	0.8/0.2	2.9/0.8	1.1/0.3
STDLoc(S+D)	0.5/0.2	0.6/0.2	0.5/0.3	0.9/0.2	0.9/0.2	0.6/0.2	1.4/0.4	0.8/0.2
Ours	0.4/0.2	0.7/0.3	0.5/0.2	0.7/0.2	0.8/0.2	0.7/0.2	1.6/0.4	0.8/0.2

4. Future Directions

We observe that our approach enables more accurate recovery of scene structure using learning-based methods. This naturally suggests that, with suitable extensions, it could be turned into a powerful framework for full scene modeling. At present, the main challenges lie in constructing robust keypoint constraints and obtaining reliable pose initialization. However, we believe these issues are tractable and are actively working on them. Preliminary scene modeling experiments based on ACEZero [7], SCRP [2], and ACE-G [8] further support the promise of this direction.

References

[1] Eduardo Arnold, Jamie Wynn, Sara Vicente, Guillermo Garcia-Hernando, Aron Monszpart, Victor Prisacariu, Daniyar Turmukhambetov, and Eric Brachmann. Map-free visual relocalization: Metric pose relative to a single image. In *ECCV*, pages 690–708. Springer, 2022. 2

[2] Wenjing Bian, Axel Barroso-Laguna, Tommaso Cavallari, Victor Adrian Prisacariu, and Eric Brachmann. Scene coordinate reconstruction priors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 25765–25776, 2025. 4

[3] Eric Brachmann and Carsten Rother. Learning less is more-6d camera localization via 3d surface regression. In *CVPR*, pages 4654–4662, 2018. 2

[4] Eric Brachmann and Carsten Rother. Visual camera relocalization from rgb and rgb-d images using dsac. *TPAMI*, 44(9):5847–5865, 2021. 2

[5] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac-differentiable ransac for camera localization. In *CVPR*, pages 6684–6692, 2017. 2

[6] Eric Brachmann, Tommaso Cavallari, and Victor Adrian Prisacariu. Accelerated coordinate encoding: Learning to relocalize in minutes using rgb and poses. In *CVPR*, pages 5044–5053, 2023. 2

[7] Eric Brachmann, Jamie Wynn, Shuai Chen, Tommaso Cavallari, Áron Monszpart, Daniyar Turmukhambetov, and Victor Adrian Prisacariu. Scene coordinate reconstruction: Posing of image collections via incremental learning of a relocalizer. *arXiv preprint arXiv:2404.14351*, 2024. 4

[8] Leonard Bruns, Axel Barroso-Laguna, Tommaso Cavallari, Aron Monszpart, Sowmya Munukutla, Victor Adrian Prisacariu, and Eric Brachmann. Ace-g: Improving generalization of scene coordinate regression through query pretraining. In *ICCV*, pages 26751–26761, 2025. 4

[9] Zhiwei Huang, Hailin Yu, Yichun Shentu, Jin Yuan, and Guofeng Zhang. From sparse to dense: Camera relocalization with scene-specific detector from feature gaussian splatting. In *CVPR*, pages 27059–27069, 2025. 2

[10] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM ToG*, 42(4), 2023. 1

[11] Xiaotian Li, Shuzhe Wang, Yi Zhao, Jakob Verbeek, and Juho Kannala. Hierarchical scene coordinate classification and regression for visual localization. In *CVPR*, pages 11983–11992, 2020. 2

[12] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1

[13] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *CVPR*, pages 2930–2937, 2013. 2

[14] Shitao Tang, Chengzhou Tang, Rui Huang, Siyu Zhu, and Ping Tan. Learning camera localization via dense scene matching. In *CVPR*, pages 1831–1841, 2021. 2

[15] Luwei Yang, Ziqian Bai, Chengzhou Tang, Honghua Li, Yasutaka Furukawa, and Ping Tan. Sanet: Scene agnostic network for camera localization. In *ICCV*, pages 42–51, 2019. 2

[16] Lei Zhou, Zixin Luo, Tianwei Shen, Jiahui Zhang, Mingmin Zhen, Yao Yao, Tian Fang, and Long Quan. Kfnet: Learning temporal camera relocalization using kalman filtering. In *CVPR*, pages 4919–4928, 2020. 2

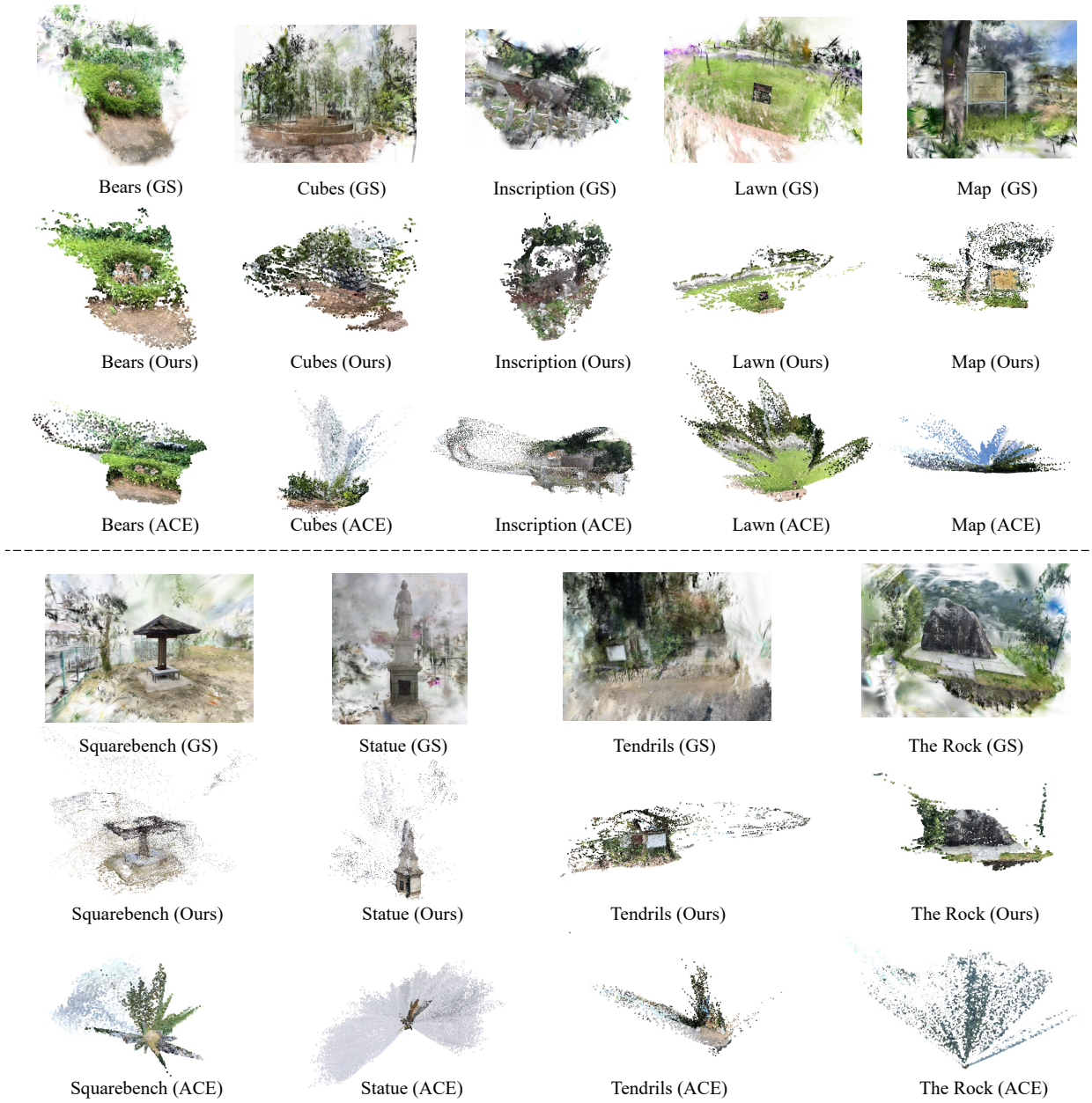

Figure 4. **Comparison of Scene Reconstruction on Wayspots.** Since our method directly regresses scene coordinates, we perform scene recovery on the Wayspots dataset after training to visualize the spatial structure learned by our network.

Algorithm 4 Stage 4: Training with Per-View Reprojection and Multi-View Bundle Loss

Require: Buffer $\mathcal{B}$; intrinsics K; camera poses $\{(R_i, t_i)\}$; weights $\lambda_\pi, \lambda_{\mathrm{B}}$
Ensure: Trained scene-coordinate regression network C_ϕ

while not converged **do**
  Sample a minibatch of tuples $(k, \mathbf{d}_k, R_i, t_i, \mathbf{u}_{ik}^{\mathrm{obs}})$ from $\mathcal{B}$
  // 4a. 3D Gaussian-splatting–enhanced observations
  Render viewpoint-consistent keypoint regions using the per-scene 3D Gaussian splatting model
  // 4b. Predict scene coordinates with C_ϕ
  Use $\mathbf{d}_k$ and rendered inputs to predict scene coordinates $\{\hat{X}_k\}$
  // 4c. Compute per-view reprojection loss L_π as in Eq.5
  $L_\pi \leftarrow 0$
  for all sample $(k, \mathbf{d}_k, R_i, t_i, \mathbf{u}_{ik}^{\mathrm{obs}})$ in minibatch **do**
    $\hat{\mathbf{u}}_{ik} \leftarrow \pi(\hat{X}_k; R_i, t_i)$
    $L_\pi \leftarrow L_\pi + \|\hat{\mathbf{u}}_{ik} - \mathbf{u}_{ik}^{\mathrm{obs}}\|_2^2$ // or Eq.5 with visibility check
  end for
  // 4d. Compute multi-view bundle loss L_{B} as in Eq.7
  Group predictions by keypoint ID s_k into $\{\mathcal{G}_j\}$ using Eq.6
  $L_{\mathrm{B}} \leftarrow 0$
  for all group $\mathcal{G}_j$ **do**
    for all $X_k \in \mathcal{G}_j$ **do**
      for all view $i \in \mathcal{V}_k$ **do**
        $\mathbf{u}_{ik} \leftarrow \pi(\hat{X}_k; R_i, t_i)$
        $L_{\mathrm{B}} \leftarrow L_{\mathrm{B}} + \|\mathbf{u}_{ik} - \mathbf{u}_{ik}^{\mathrm{obs}}\|_2^2$
      end for
    end for
  end for
  // 4e. Total loss as in Eq.8
  $L_{\mathrm{total}} \leftarrow \lambda_\pi L_\pi + \lambda_{\mathrm{B}} L_{\mathrm{B}}$
  Backpropagate L_{total} and update C_ϕ (and optionally camera poses)
end while