

Scene Grounding In the Wild

Supplementary Material

We refer readers to the interactive visualizations at the accompanying InteractiveViewer/index.html that show results for all presented baseline models (before and after our inverse optimization scheme) on the *WikiEarth* test set.

1. Additional Results and Comparisons

1.1. Additional Quantitative Results

In addition to the averaged ΔR , ΔT reported in the main paper (Table 1), in Tables 1, 2, 3, we report a per meta-image performance breakdown for all the initializations. From this breakdown, we observe that our method successfully registers meta-images where the baseline exhibits large $\Delta R_{\mathcal{I}}$, $\Delta T_{\mathcal{I}}$ errors, such as lines 8, 13 and 14 in the COLMAP baseline table. However, there are cases where both our method and the baseline fail to register the images, as seen in lines 10, 31.

ID	WikiScenes ID	Name	Meta Image ID	Baseline $\Delta T_{\mathcal{I}}$	Baseline $\Delta R_{\mathcal{I}}$	Our $\Delta T_{\mathcal{I}}$	Our $\Delta R_{\mathcal{I}}$
1	0	Milano Cathedral	30	0.216	4.114	0.348	4.822
2	0	Milano Cathedral	30	0.198	4.019	0.293	5.610
3	10	Sanfrancisco Cathedral	0	0.329	6.224	0.157	2.562
4	18	Lisbon Cathedral	0	0.064	1.835	0.135	1.973
5	20	Brussels Cathedral	16	0.008	1.687	0.024	0.518
6	20	Brussels Cathedral	15	0.051	1.669	0.076	0.924
7	20	Brussels Cathedral	0	0.085	0.631	0.105	0.877
8	20	Brussels Cathedral	13	0.182	15.723	0.063	1.228
9	20	Brussels Cathedral	14	0.047	1.708	0.076	0.910
10	21	Salamanca Cathedral	1	0.516	11.609	0.481	9.728
11	27	St John the Divine Cathedral	0	0.005	0.515	0.033	1.341
12	36	Burgos Cathedral	10	0.019	0.921	0.189	3.000
13	37	Aachen Cathedral	7	0.086	41.417	0.146	3.944
14	39	Basel Minster	4	0.558	24.068	0.212	3.742
15	43	Oviedo Cathedral	1	0.006	0.255	0.089	1.385
16	46	Palermo Cathedral	1	0.026	0.696	0.089	1.320
17	46	Palermo Cathedral	0	0.018	0.379	0.044	0.716
18	50	Wells Cathedral	1	0.219	3.072	0.095	2.138
19	52	Lincoln Cathedral	2	0.024	0.493	0.051	0.516
20	52	Lincoln Cathedral	11	0.017	1.076	0.041	1.368
21	53	Monreale Cathedral	0	0.007	0.518	0.032	0.476
22	54	Rouen Cathedral	0	0.021	1.099	0.082	2.622
23	55	Geneva Cathedral	2	0.026	3.808	0.252	6.617
24	56	Ulm Minster	1	0.225	4.468	0.189	3.965
25	61	Bordeaux Cathedral	3	0.011	0.254	0.089	2.713
26	61	Bordeaux Cathedral	0	0.291	8.810	0.095	2.599
27	75	Murcia Cathedral	0	0.051	1.398	0.039	0.650
28	85	Metz Cathedral	2	0.159	2.883	0.067	0.480
29	85	Metz Cathedral	1	0.143	5.196	0.098	1.214
30	90	Avila Cathedral	1	0.035	1.325	0.060	2.437
31	93	salisbury cathedral	4	0.173	6.771	0.215	6.537
32	97	Napoli Cathedral	1	0.018	0.941	0.023	0.476

Table 1. **Performance Breakdown Per Meta-Image.** Performance of our method (initialized with COLMAP) and the COLMAP baseline per meta-image, considering only meta-images where the baseline did not fail.

As mentioned in the main paper, there are cases where the baselines fails and does not output any transformation. Empty lines in the results tables indicate baseline failures. Upon a closer look of the COLMAP and SP+LG baseline failures, we found that it fails because it is unable to register the minimum of three images to the reference model, which is required for the global transform estimation.

Additionally Tab. 4 Tab. 5 present the MTA and O% scores across various thresholds. These tables show our method consistently outperforms the baseline across all

ID	WikiScenes ID	Name	Meta Image ID	Baseline $\Delta T_{\mathcal{I}}$	Baseline $\Delta R_{\mathcal{I}}$	Our $\Delta T_{\mathcal{I}}$	Our $\Delta R_{\mathcal{I}}$
1	0	Milano Cathedral	30	0.228	2.874	0.360	4.995
2	0	Milano Cathedral	30	0.197	4.102	0.299	5.691
3	10	Sanfrancisco Cathedral	0	0.237	4.710	0.159	2.551
4	18	Lisbon Cathedral	0	0.084	1.211	0.131	1.845
5	20	Brussels Cathedral	16	0.016	0.394	0.024	0.519
6	20	Brussels Cathedral	15	0.076	0.953	0.075	0.897
7	20	Brussels Cathedral	0	0.085	0.502	0.103	0.827
8	20	Brussels Cathedral	13	0.032	0.599	0.057	1.144
9	20	Brussels Cathedral	14	0.071	0.926	0.075	0.901
10	21	Salamanca Cathedral	1	0.874	21.297	0.870	21.263
11	27	St John the Divine Cathedral	0	0.005	0.224	0.034	1.372
12	36	Burgos Cathedral	10	0.019	0.901	0.197	3.127
13	37	Aachen Cathedral	7	0.059	1.705	0.142	3.807
14	39	Basel Minster	4	0.643	17.056	0.186	2.945
15	43	Oviedo Cathedral	1	0.006	0.181	0.087	1.340
16	46	Palermo Cathedral	1	0.027	0.661	0.085	1.297
17	46	Palermo Cathedral	0	0.018	0.325	0.048	0.821
18	50	Wells Cathedral	1	0.011	0.268	0.096	2.181
19	52	Lincoln Cathedral	2	0.025	0.390	0.050	0.501
20	52	Lincoln Cathedral	11				
21	53	Monreale Cathedral	0	0.006	0.179	0.033	0.508
22	54	Rouen Cathedral	0	0.034	1.370	0.081	2.592
23	55	Geneva Cathedral	2	0.018	0.748	0.015	0.709
24	56	Ulm Minster	1	0.224	4.240	0.218	4.372
25	61	Bordeaux Cathedral	3	0.012	0.303	0.088	2.677
26	61	Bordeaux Cathedral	0	0.289	8.118	0.100	2.586
27	75	Murcia Cathedral	0	0.051	1.298	0.041	0.687
28	85	Metz Cathedral	2	0.159	2.882	0.066	0.486
29	85	Metz Cathedral	1	0.065	0.704	0.098	1.227
30	90	Avila Cathedral	1	0.036	1.349	0.059	2.428
31	93	salisbury cathedral	4	0.214	7.259	0.214	6.558
32	97	Napoli Cathedral	1	0.020	0.862	0.025	0.473

Table 2. **Performance Breakdown Per Meta-Image gDLS+++ initialization.** Performance of our method (initialized with gDLS+++ and the gDLS+++ baseline per meta-image

ID	WikiScenes ID	Name	Meta Image ID	Baseline $\Delta T_{\mathcal{I}}$	Baseline $\Delta R_{\mathcal{I}}$	Our $\Delta T_{\mathcal{I}}$	Our $\Delta R_{\mathcal{I}}$
1	0	Milano Cathedral	30				
2	0	Milano Cathedral	30				
3	10	Sanfrancisco Cathedral	0	0.146	3.367	0.161	2.612
4	18	Lisbon Cathedral	0	0.088	1.659	0.135	1.985
5	20	Brussels Cathedral	16	0.019	1.250	0.030	1.521
6	20	Brussels Cathedral	15	0.059	3.376	0.087	8.217
7	20	Brussels Cathedral	0	0.083	0.476	0.105	0.848
8	20	Brussels Cathedral	13	0.040	2.145	0.071	0.652
9	20	Brussels Cathedral	14	0.104	11.597	0.075	0.908
10	21	Salamanca Cathedral	1	2.005	17.949	2.006	17.997
11	27	St John the Divine Cathedral	0	0.011	0.478	0.033	1.276
12	36	Burgos Cathedral	10	2.081	10.503	2.124	9.823
13	37	Aachen Cathedral	7	0.128	5.099	0.141	3.936
14	39	Basel Minster	4	0.156	4.468	0.211	4.296
15	43	Oviedo Cathedral	1				
16	46	Palermo Cathedral	1	0.026	1.067	0.084	1.233
17	46	Palermo Cathedral	0	0.034	0.640	0.048	0.806
18	50	Wells Cathedral	1	0.021	0.510	0.088	2.079
19	52	Lincoln Cathedral	2				
20	52	Lincoln Cathedral	11				
21	53	Monreale Cathedral	0	0.025	0.578	0.032	0.464
22	54	Rouen Cathedral	0	0.073	1.881	0.074	2.501
23	55	Geneva Cathedral	2	0.075	2.509	0.074	2.597
24	56	Ulm Minster	1	0.278	2.182	0.185	3.752
25	61	Bordeaux Cathedral	3	0.070	3.517	0.094	2.715
26	61	Bordeaux Cathedral	0	0.569	4.976	0.102	2.616
27	75	Murcia Cathedral	0	0.056	0.926	0.045	0.696
28	85	Metz Cathedral	2	0.369	5.801	0.086	0.636
29	85	Metz Cathedral	1	0.072	1.250	0.093	1.003
30	90	Avila Cathedral	1	0.098	4.523	0.074	2.947
31	93	salisbury cathedral	4	0.186	7.064	0.214	6.530
32	97	Napoli Cathedral	1	0.021	0.922	0.024	0.472

Table 3. **Performance Breakdown Per Meta-Image SP+LG initialization.** Performance of our method (initialized with SP+LG) and the SP+LG baseline per meta-image

Methods	MTA _(3,0.1)	MTA _(7,0.1)	MTA _(10,0.1)	MTA _(5,0.09)	MTA _(5,0.15)	MTA _(5,0.2)
Ours (Colmap init)	62	62	62	53	72	81
Colmap Baseline	56	59	59	59	59	66

Table 4. **Performance over Different MTA Thresholds.** Above, we report MTA scores over different threshold values for both the baseline and our model. Note that $MTA_{(r,t)}$ considers an image as accurately registered if $\Delta R < r$ and $\Delta T < t$.

thresholds, considering both the MTA and O%.

Methods	$O\%_{(10,0.3)}$	$O\%_{(10,0.4)}$	$O\%_{(10,0.6)}$	$O\%_{(15,0.5)}$	$O\%_{(17,0.5)}$	$O\%_{(20,0.5)}$
Ours (Colmap init)	6	3	0	0	0	0
Colmap Baseline	16	12	12	12	9	9

Table 5. **Performance over Different O% Thresholds.** Above, we report outlier scores over different threshold values for both the baseline and our model. Note that $O\%_{(r,t)}$ considers an image as an outlier if $\Delta R > r$ or $\Delta T > t$.

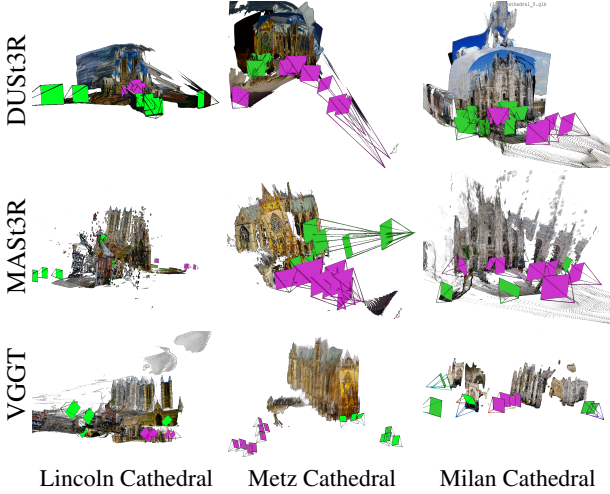


Figure 1. **Reconstructions of Feed-Forward Models.** We visualize three reconstructions obtained by running DUST3R [7] MAST3R [2] and VGGT [6] over images sampled from two meta-images (visualized in **green** and **purple**). As illustrated above, all methods fail to reconstruct the Milan and Lincoln Cathedral, showing either broken or overlapping meta-images, while these cameras capture non-overlapping regions as seen in the ground truth reconstructions (see illustration in the main paper).

1.2. Additional Qualitative Comparison

In the paper, we qualitatively show that π^3 struggles to register meta-image to reference model, illustrating that existing sparse reconstruction approaches cannot address the task we propose. In Fig. 2, we present typical failures cases of VGGT when registering a meta-image to a reference model. In Fig. 1, we qualitatively show failure cases of the feed-forward methods when registering non-overlapping meta-images.

In Fig. 3 we show additional qualitative comparison to the COLMAP baseline using a reference model created from drone videos. In Fig. 4 we show the results of our method on the IMC-PT dataset compared to the COLMAP baseline. In Fig. 5 we show additional qualitative comparison to the COLMAP baseline.

1.3. Method Analysis

As mentioned in the main paper, to handle image outliers we use robust optimization method (LTS [4]). In each optimization iteration the LTS ignores images with loss higher

than the median. We visualize the ignored images distribution in Fig. 6. As discussed in Section 5.4 of the main paper, the histograms reveal that most images are either constantly ignored (in the rightmost bin) or never ignored (in the leftmost bin). However, some image fall into the middle bins where they are sometimes ignored and sometimes not. The histograms further illustrate that our robust optimization scheme provides a soft selection mechanism enabling stable convergence.

Additionally, we visualize images from several bins. These demonstrate that the images that are always ignored by our method are indeed outliers. Specifically, these are images with occlusion and images that resides behind floaters in the reference model.

1.4. Comparison with Feed-Forward 3D Models

In the paper, we compare our method with three baselines: VGGT, MAST3R, π^3 and DUST3R. We ran MAST3R and DUST3R using the master-sfm code provided in their official repository, we ran π^3 from the official repository and we ran VGGT using the demo-colmap setup from its official repository. Due to GPU memory limitations on our A5000 GPU, we were unable to run these baseline (except π^3) methods with more than 45 input images. We were able to run π^3 with up to 180 images.

For the meta-to-meta experiment, we sampled 22 random images from each meta image and ran the experiment 5 times.

For the meta-to-reference experiment, we sampled 35 images from the reference model and 10 random images from the meta model. We chose to sample 35 images from the reference model to ensure sufficient coverage of the entire scene. We did not sample random images from the reference model because random sampling often failed to cover the scene, causing the Feed-Forward methods to fail. Instead, we selected evenly spaced images, since the images were captured sequentially. We repeated this experiment 5 times, each with a different image offset.

2. Limitations

While our method is not specifically designed for single-shot scenarios, we evaluate its reliability with fewer images per meta-image in Fig. 7 (left). We evaluate performance by randomly sub-sampling subsets of varying sizes from each meta-image, reporting the average error across five independent samples. Performance drops over very small meta-images due to an insufficient number of informative images to guide our alignment scheme. However, results remain largely stable for small collections containing at least six images. Furthermore, our method is challenged by very noisy initializations. As illustrated in Fig. 7 (right), adding noise to the rotation parameters leads to a significant drop in performance.

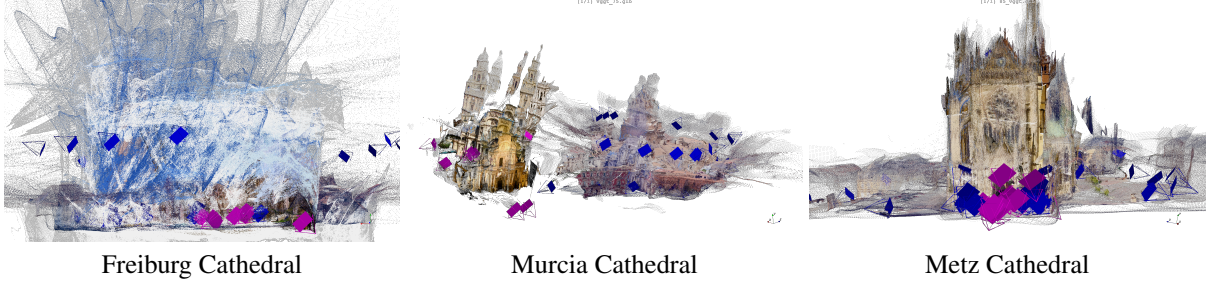


Figure 2. **Aligning a meta-image to a reference model with VGGT.** Meta-image cameras are visualized in purple, while Google Earth images from the reference model are in blue. As illustrated above, VGGT failed to register the Murcia meta-image (*i.e.*, the output contains two distinct regions, one for the Internet images and one for the Google Earth images) and also failed to reconstruct the reference model in the Freiburg Cathedral. For the Metz Cathedral, VGGT produced a slightly misaligned registration, as evident by the ghost structures near the top of the building.

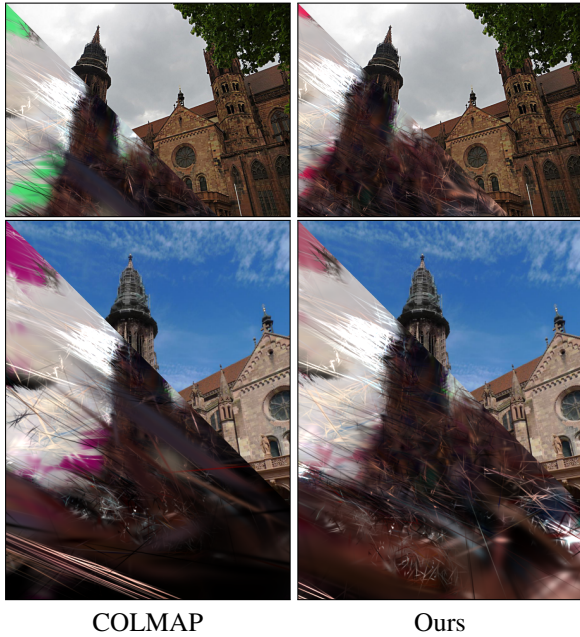


Figure 3. **Drone Reference model - Additional examples** Results using a reference model from drone video frames are depicted above. The drone videos of the Freiburg Cathedral are taken from Youtube. As illustrated our approach significantly improves the alignment, in comparison to the COLMAP baseline, which serves as our initialization.

3. Implementation Details

3.1. The reference model

First we extract DINOv2 [3] dense features per rendered landmark image from Google Earth Studio. We resize each image to 1400X1400 and then use the pretrained backbone *dinov2_vits14*, which outputs dense feature map 100X100. We chose DINOv2 with embedding size of 384. We use the DINO implementation *facebookresearch/dinov2* in Github.

We use the landmark images rendered from Google

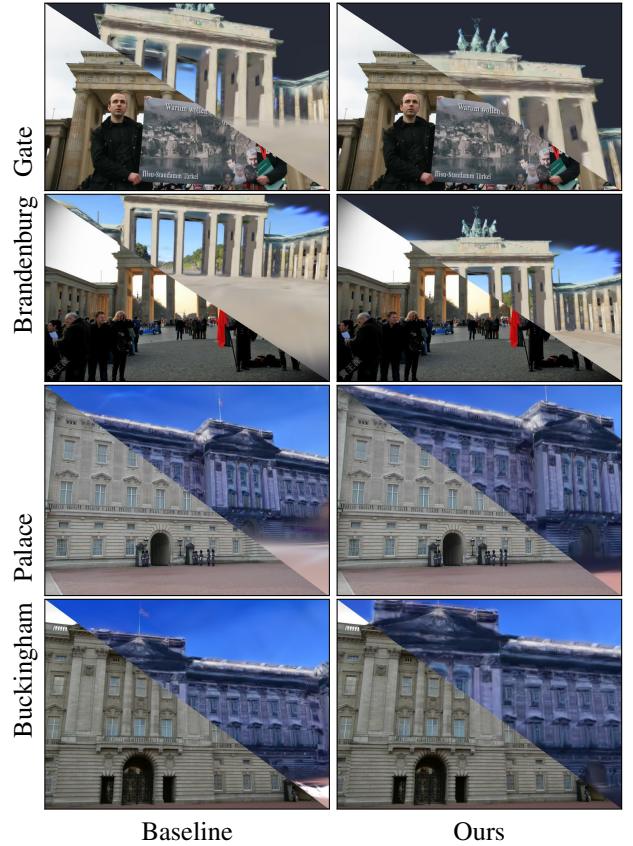


Figure 4. **Qualitative Comparison on Different Scene Types** A visualization of the alignment results for our method and the COLMAP baseline. The scenes are taken from the IMC-PT [1] dataset.

Earth, the COLMAP model of those images from the benchmark, and the extracted DINOv2 feature to train 3DGS. We follow the implementation of Feature 3DGS[9]. We chose feature vector per Gaussian with size 128. We apply the Speedup Model, a Conv2d network with input 128 output

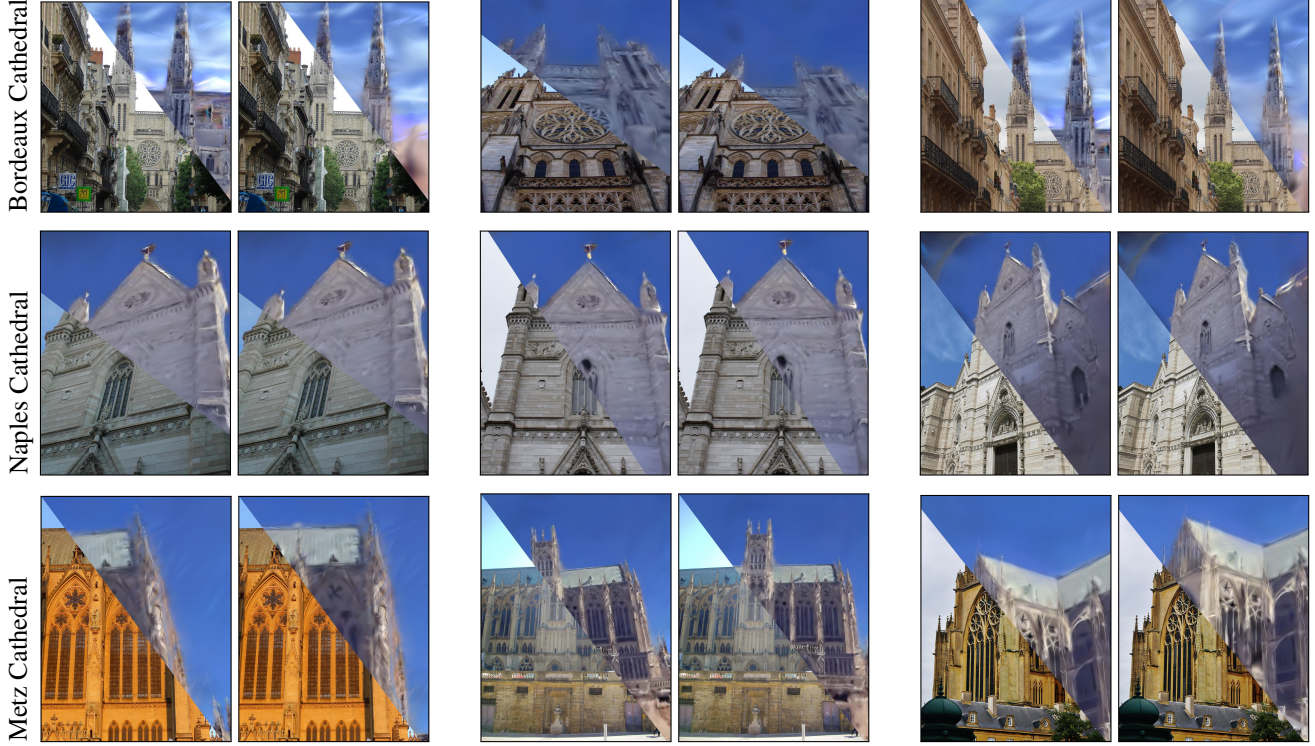


Figure 5. **Additional Qualitative Comparison:** A visualization of the alignment results for our method and the COLMAP baseline. Each image shows the ground truth in the lower half and the rendered image from the reference model \mathcal{M} after alignment in the top half. As demonstrated, our inverse optimization-based approach predicts precise transformations, even in the presence of challenging, inaccurate initializations.

384 and kernel size 1 to decode the features of the Gaussian to the DINOv2 Features. We integrated the implementation of Feature3DGS to Nerf Studio, specifically to Splatfacto - the Gaussian Splatting implementation in Nerf Studio. In each training iteration we choose one image, render its feature (size 128) pass it through the speedup module which translated it to size 384. Then we use bilinear interpolation to resize the rendered feature image to 100X100 to match the DINOv2 features which was extracted beforehand. Similarly to [9], our Loss function is:

$$L = L_{rgb} + \left| F_t(I) - F_s(\hat{I}) \right| \quad (1)$$

Where L_{rgb} is the regular 3DGS, I is the Image, $F_t(I)$ is the extracted features of the image in the pre-process, and $F_s(\hat{I})$ is the rendered images after the a pass through the speed up module and the bilinear interpolation.

For the 3DGS parameters we use the parameters of the Spaltfacto method in NerfStudio [5]. We use Adam Optimizer for the gaussian feature vectors with lr=0.05, eps=1e-15, and exponential decay scheduler with lr-final=1.6e-6. For the Feature Speedup Module we use Adam optimizer with lr=0.001, eps=1e-15. We train the model for 12000 steps.

3.2. Initialization

We use COLMAP, SP+LG and gDLS+++ to initialize the global transformation of meta-image \mathcal{I} . To find the global transformation using COLMAP we first register the images in the meta-image one by one. We give COLMAP as an input the landmark model we previously built from Google Earth Studio images. We fix the input model (using the flag fix-existing-images) when running the COLMAP exhaustive matcher and the mapper. To find the global best transform we align the meta-image to the registered images using COLMAP model aligner. COLMAP model aligner uses point set registration and RANSAC. For the COLMAP aligner we used alignment-max-error=3, alignment-type="custom" and ref-is-gps=0. For the SP+LG initialization we perform the same steps as the COLMAP initialization, but replacing the feature extractor from SIFT to SuperPoint and the feature matching to Light Glue.

3.3. Registration Implementation Details

For the registration vector we used Adam Optimizer with lr=1e-3, eps=1e-8.

We noticed that in some 3DGS models there were floaters around the ground, which impede the convergence

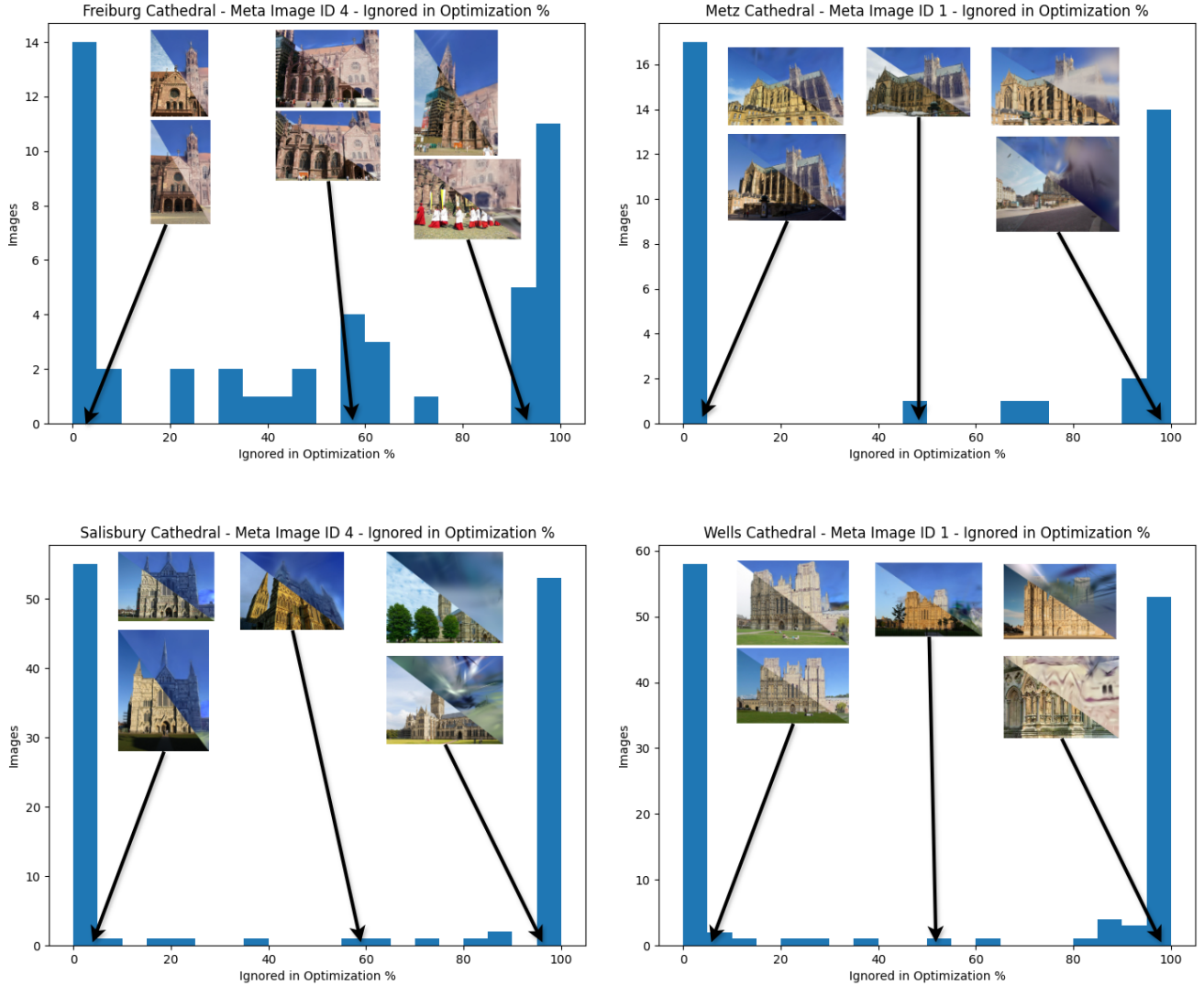


Figure 6. **Additional Analysis of our Robust Optimization Framework.** Our method uses LTS to ignore the images with loss values that are higher than the median image loss. In the figure, we show a sample of images in several bins. The image below the diagonal are the real-world Internet images, and above is the rendered image from the reference model (rendered at the end of the optimization). As illustrated by the rightmost bin, images there are typically outliers, *i.e.*, images with occlusions and images that resides behind floaters in the reference model. Further analysis is provided in Sec. 1.3.

of our registration in some cases. To mitigate it we set the near plane of all the cameras 0.7, so the cameras will not render the gaussians near them, especially the floaters on the ground.

3.4. Runtime

The optimization (creating the reference model, performed once per reference model) takes roughly 15 minutes on a Nvidia A5000 GPU. The registration (of the meta image to the reference model) is performed over 12000 steps, taking about 5 minutes on a Nvidia A5000 GPU.

4. The WikiEarth Benchmark

We rendered images around each landmark using Google Earth Studio, the camera trajectories for each landmark will be published with the benchmark. The Google Earth Studio rendering UI is presented at Fig. 8.

After rendering the images, we create a COLMAP using the rendered images of the landmark from Google Earth Studio. We use COLMAP spatial matcher, utilizing the GPS coordinates saved in the rendered image by Google Earth Studio and we configure the mapper with the flag "ig-

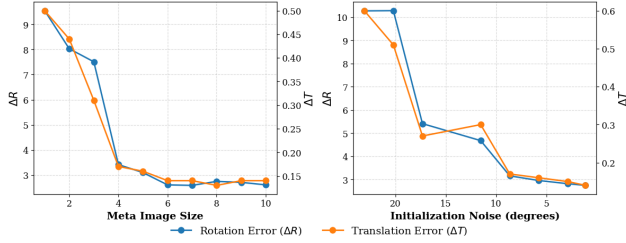


Figure 7. **Robustness Analysis.** Average errors ΔR and ΔT across the benchmark as a function of the meta-image size and the initialization noise (rotation). The graphs indicate that the error increases with smaller meta-images, reaching a plateau at a size of approximately 6. Furthermore, the initialization graph demonstrate that the method aligns the images successfully once the noise is below a specific threshold (10°) for each of the rotation parameters; see Section 2 for additional details.

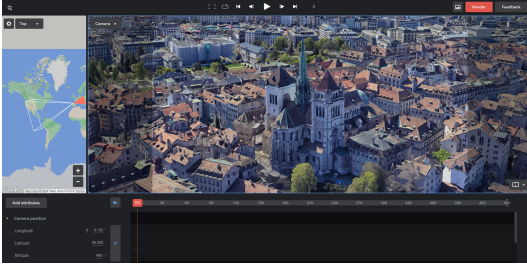


Figure 8. **Google Earth Studio UI:** Screenshot of Google Earth Studio, showing the 3D model of the Geneva Cathedral. For each landmark we create a camera trajectory and rendered the images on the trajectory using the program

none_watermarks”.

We aligned this model images from the WikiScenes dataset, we chose only images in the exterior category for each landmark. The images are mostly not registered correctly with the default COLMAP parameters, for each landmark we manually found the best parameters, presented in Tab. 6. Then we manually removed images that were not registered correctly. The benchmark is described in Tab. 7.

To compare meta-image alignment to the benchmark, the meta-image camera intrinsics must match the intrinsic on the benchmark. To enable evaluation, we forced the intrinsics of the benchmark’s cameras on the meta images by rebuilding the meta image.

References

- [1] Yuhe Jin, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image matching across wide baselines: From paper to practice. *International Journal of Computer Vision*, 129(2):517–547, 2020. 3
- [2] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r, 2024. 2
- [3] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo,

WikiScenes ID	Name	Required Matches
0	Milano Cathedral	13
10	San Francis	13
18	Lisbon Cathedral	30
20	Brussels Cathedral	12
21	Salamanca Cathedral	17
27	St John the Diving Cathedral	15
36	Burgos Cathedral	17
37	Aachen Cathedral	13
39	Freiburg Cathedral	15
43	Oveido Cathedral	13
46	Palermo Cathedral	13
50	Wells Cathedral	13
52	Lincoln Cathedral	13
53	Monreale Cathedral	13
54	Rouen Cathedral	19
55	Geneva Cathedral	14
61	Bordeaux Cathedral	14
75	Murcia Cathedral	14
85	Metz Cathedral	18
90	ávila Cathedral	15
93	Salisbury Cathedral	30
97	Napoli Cathedral	25

Table 6. **COLMAP Parameters for WikiEarth Benchmark Creation.** We run COLMAP mapper with two-view tracks, increased triangulation max transitivity (3), increased absolute pose maximal error (36), and increased bundle adjustment maximal refinement range (0.0015). Additionally we run the mapper with varying number of Required Matches described in the table.

- Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024. 3
- [4] Peter J. Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, 79(388):871–880, 1984. 2
- [5] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, David Mcallister, Justin Kerr, and Angjoo Kanazawa. Nerfstudio: A modular framework for neural radiance field development. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings*, page 1–12. ACM, 2023. 4
- [6] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 2
- [7] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy, 2024. 2
- [8] Xiaoshi Wu, Hadar Averbuch-Elor, Jin Sun, and Noah Snavely. Towers of babel: Combining images, language, and 3d geometry for learning multimodal vision. In *Proceedings*

WikiScenes ID	Name	Meta Image ID	Meta Image Size
0	Milano Cathedral	3	19
0	Milano Cathedral	30	713
10	Sanfrancisco Cathedral	0	21
18	Lisbon Cathedral	0	289
20	Brussels Cathedral	16	8
20	Brussels Cathedral	15	12
20	Brussels Cathedral	0	168
20	Brussels Cathedral	13	12
20	Brussels Cathedral	14	12
21	Salamanca Cathedral	1	342
27	St John the Divine Cathedral	0	33
36	Burgos Cathedral	10	24
37	Aachen Cathedral	7	23
39	Basel Minster	4	53
43	Oviedo Cathedral	1	73
46	Palermo Cathedral	1	33
46	Palermo Cathedral	0	188
50	Wells Cathedral	1	141
52	Lincoln Cathedral	2	125
52	Lincoln Cathedral	11	14
53	Monreale Cathedral	0	35
54	Rouen Cathedral	0	184
55	Geneva Cathedral	2	26
56	Ulm Minster	1	122
61	Bordeaux Cathedral	3	20
61	Bordeaux Cathedral	0	39
75	Murcia Cathedral	0	52
85	Metz Cathedral	2	59
85	Metz Cathedral	1	40
90	Avila Cathedral	1	110
93	salisbury cathedral	4	131
97	Napoli Cathedral	1	43

Table 7. **Benchmark Scenes.** The *WikiEarth* benchmark consists of 32 meta-images from 23 different landmarks found in the WikiScenes [8] dataset, as detailed above.

of the *IEEE/CVF International Conference on Computer Vision*, pages 428–437, 2021. 7

- [9] Shijie Zhou, Haoran Chang, Sicheng Jiang, Zhiwen Fan, Zehao Zhu, Dejia Xu, Pradyumna Chari, Suya You, Zhangyang Wang, and Achuta Kadambi. Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21676–21685, 2024. 3, 4