Light3R-SfM: Towards Feed-forward Structure-from-Motion
- Supplementary Material -

Sven Elflein[1,2,3] Qunjie Zhou[1] Laura Leal-Taixé[1]

[1]NVIDIA [2]Vector Institute [3]University of Toronto

In this supplementary document, we provide implementation details in Appendix A, additional evaluations on Tanks&Temples for pose estimation (Appendix B) and Waymo Open Dataset for 3D reconstruction (Appendix C). Next, we test our method on disconnected image collections, i.e., collections of images of different scenes, in Appendix D and investigate robustness against drift in Appendix E. Furthermore, we conduct more ablation studies to validate our model design in Appendix F, followed by qualitative visualizations in Appendix G.

A. Implementation Details

Model. Our model adopts the same encoder and decoder architecture as DUSt3R, *i.e.*, the VIT-L image encoder and the two pointmap decoding regression heads parameterized by VIT-B [3]. For global alignment, we use $L = 4$ blocks. `Self` and `Cross` are implemented as vanilla self- and cross-attention layers [17] with 8 attention heads and pre-normalization. Their feature dimensionality is the same as the VIT-L encoder dimension, *i.e.*, 1024.

Training. We train our model on four datasets: Waymo Open Dataset [15], CO3Dv2 [12], MegaDepth [9], and TartanAir [21]. For training, we sample graphs of $N = 8$ images based on pairwise scores proposed in CroCo [22] and a greedy algorithm which iteratively adds additional images with maximum viewpoint angle difference w.r.t. all images already in the set, until the desired number of images is reached. Images are resized such that their longer side has length 512 and then center cropped such that the shorter side is in $\{384, 336, 288, 256, 160\}$ leading to different aspect ratios for training. Further, we apply color jitter augmentation. We initialize our model encoder and decoder using MASt3R pretrained weights and train for 100,000 iterations with batch size 8 (each batch element corresponds to one graph of images) using AdamW [10] with learning rate 10^{-6} and weight decay 5×10^{-4} on 8 NVIDIA A100-80GB GPUs. The model on small resolutions (using 224×224) is trained on 16 NVIDIA V100-32 GPUs with per-GPU batch size of 2, resulting in overall batch size of 32. We scale the learning rale linearly with batch size.

Inference. At test time, we extract the global camera pose from the pointmaps in global reference frame X^i and their corresponding confidence maps C^i. We follow Wang et al. [20] and first estimate the focal length with a robust estimator [23] and then proceed to extract the pose with RANSAC-PnP [5, 16] from points with their corresponding confidence in C^i larger than a threshold. By default we use a threshold of 3, or the 90%-quantile if all confidences fall below the threshold.

To reduce regression noise, we further symmetrize the edges during inference and combine the pointmap predictions using a confidence-weighted average. In detail, we decode the symmetric edge (j, i), now predicting in the reference frame of image j, for every edge $(i, j) \in E_{\texttt{SPT}}$, extract the pairwise pose using Procrustes as described in the main paper, then apply the transformation to the output pointmaps $X^{j,i}$, $X^{j,j}$ to obtain $\tilde{X}^{i,i}$ and $\tilde{X}^{i,j}$ respectively. We then compute the confidence-weighted average for the pointmaps of the edge (i, j) we are interested in. Here, we introduce the computation to combine $X^{i,i}$ and $\tilde{X}^{i,i}$ but it applies symmetrically to $X^{i,j}$. First, we compute weight from the confidences $C^{i,i}$ corresponding to edge (i, j) and $C^{j,i}$ from edge (j, i) as

$$G^{i,i}_{u,v} = \frac{\log C^{i,i}_{u,v}}{\log C^{i,i}_{u,v} + \log C^{j,i}_{u,v}}$$

where $u \in \{1, \ldots, W\}, v \in \{1, \ldots, H\}$ are indexing into the confidence-/pointmaps. Note that the confidence maps correspond to the same image fed in different position to the pairwise decoder. We then compute the average-weighted pointmap as

$$X^{i,i}_{u,v} := (G^{i,i}_{u,v}) X^{i,i}_{u,v} \cdot (1 - G^{i,i}_{u,v}) \tilde{X}^{i,i}_{u,v}$$

incorporating information from the decoder evaluation of the symmetric edge, thus refining the pointmap.

B. Additional Details on Tanks & Temples [7]

Runtime evaluation. For completeness, we report the per-scene reconstruction runtime for all baseline methods in

Tab. 7. For fair comparison, we run other methods using their open-source implementation with default parameters provided with the code on the same base system with 10 CPU cores, 64GB system memory, and one NVIDIA V100 GPU with 32GB VRAM. For MASt3R-SfM [8], we adopt the hyper-parameters reported in the paper. We have to do specific adjustment for VGGSfM [19] to fit the GPU memory budget where we follow their suggestions [1] and reduce `max_points_num` to $40,960$ and `max_tri_points_num` to $204,800$, *i.e.*, $1/4$ their original values. However, this still leads to out-of-memory errors when evaluating on most of the full sequences and some of the 200-image sequences supposedly due to excessive amount of detected keypoints, and thus we do not report the runtime results in these situations.

Pose accuracy evaluation. In the main paper, we report pose accuracy at tight error threshold of 5°. In Fig. 1, we provide a more complete overview of the model performance at other thresholds by plotting the pose accuracy as a function of the error threshold for both relative rotation and translation errors. We observe a gap at tight thresholds between *feed-forward* approaches (Light3R-SfM, Spann3R [18]) and *optimization-based* approaches, however, this gap rapidly shrinks for our method when moving towards looser thresholds, while Spann3R is consistently worse. This suggests that Light3R-SfM is generally able to locate the correct positions and orientations of cameras while struggling to regress the exact values which is more easily achieved via optimization.

For some downstream applications that perform pose refinement, *e.g.*, novel-view synthesis via Gaussian splatting [6], these coarse poses might already be sufficient and can directly enjoy the significant speed-ups of up to 198× of our method. Further, these results suggest that a small optimization stage on top of the regressed outputs, converging fast due to good initialization, could significantly increase performance at tight thresholds. We leave investigation into this direction to future work.

C. Evaluation on 3D Reconstruction

We further evaluate our method on 3D reconstruction using Waymo Open Dataset [15]. We evaluate the quality of the global predicted point cloud per scene by computing the Chamfer distance [1] w.r.t. the sparse lidar ground-truth point cloud. For this, we find the nearest neighbor for every ground truth point and compute the euclidean distance, then compute the average. We compare ourselves to Spann3R and MASt3R-SfM, as well as a variant of our method without latent global alignment. In Fig. 3, we report the cumulative distribution function of per-scene reconstruction errors as measured by the Chamfer distance.

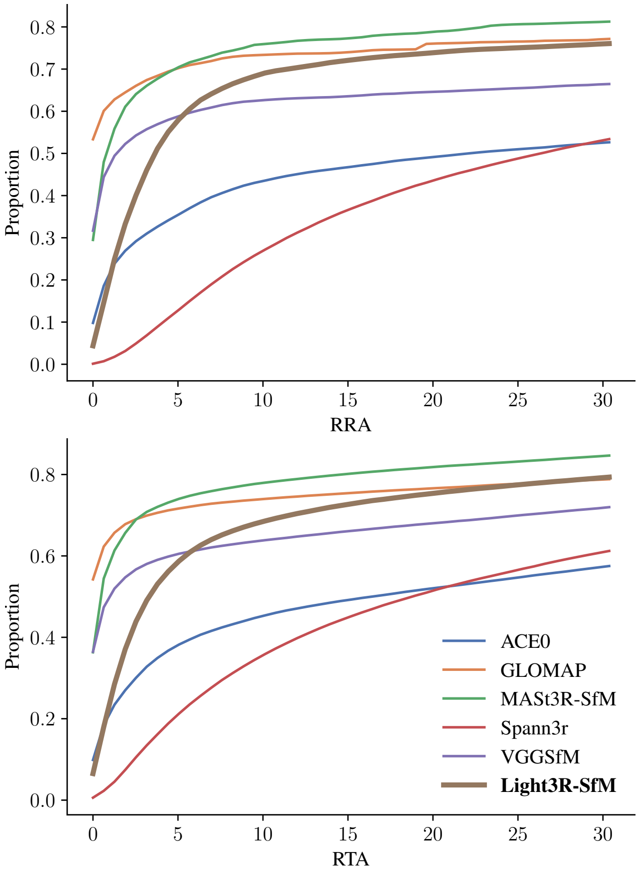

Figure 1. **CDF of pose errors on 100-view Tanks&Temples scenes.**

We show that our methods with and without latent global alignment are both able to largely outperform Spann3R, producing point cloud with smaller reconstruction errors for most of the scenes. It confirms the limitation of Spann3R in handling non-object-centric, natural scenes. We further highlight that our method with latent global alignment module is significantly better than the baseline without it (*w/o lat.align*), validating its effectiveness to ensure global consistency across pairwise pointmaps, even for the long, forward-moving trajectories.

Compared to the *optimization-based* MASt3R-SfM, Light3R-SfM manages to produce a subset of reconstructions with lower reconstruction errors. However, there is also a proportion of scenes where our method falls behind. After investigation, we find that these scenes contain many dynamic objects. Light3R-SfM was mostly trained on static scenes, and thus often assigns confidence to portions of the pointmap that are dynamic resulting in wrong pairwise pose estimates, affecting global accumulation, and thus degrading global reconstructions. For illustration, we visualize the confidence map for such a dynamic scene in Fig. 2. MASt3R-SfM, despite building on top of MASt3R as well,

[1] https://github.com/facebookresearch/vggsfm/blob/main/README.md

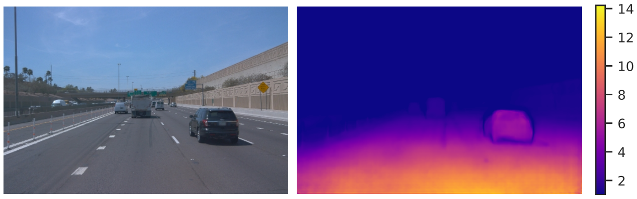

Figure 2. Global confidence map (right) produced by Light3R-SfM for an image of a sequence containing dynamic objects (left).

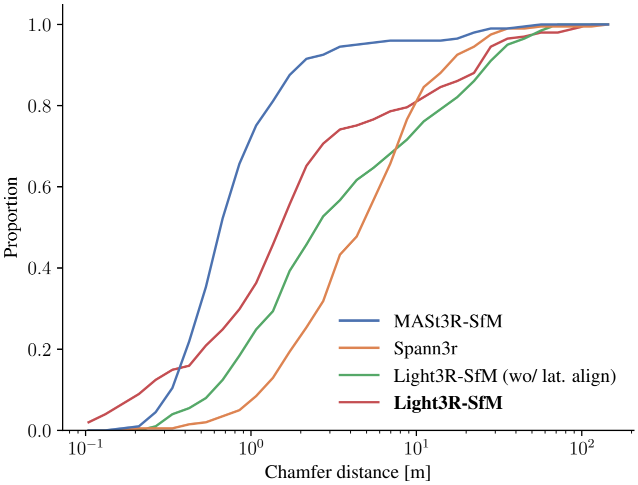

Figure 3. **CDF of per-scene 3D reconstruction errors.**

	7scenes		Waymo	
	Acc. ↓	Comp. ↓	Acc. ↓	Comp. ↓
Spann3R	**0.0108**	**0.0104**	7.114	23.486
DUSt3R (global opt.)	0.0133	0.0108	OOM	
Ours (w/o latent align.)	0.0111	0.0241	5.942	9.506
Ours	0.0111	0.0196	**5.065**	**2.620**

Table 1. 3D reconstruction metrics on 7scenes [14] and Waymo Open Dataset [15]

performs better in these situations as erroneous correspondences on dynamic objects are discounted by a robust error function during optimization. We believe Light3R-SfM will be able to handle these scenes by training on more diverse datasets containing dynamic objects, as the global supervision will encourage low confidence for dynamic parts of the image.

We report median accuracy and completion metrics on 7Scenes [14] few-view setting following Spann3R in Tab. 1. All methods are evaluated on 224 × 224 resolution images for a fair comparison.

As shown in Tab. 1, Light3R-SfM produces competitive results on 7scenes and we observe that latent global alignment improves 3D reconstruction. We believe that scaling compute and incorporating indoor training data will further close this gap. On Waymo Open Dataset, our method significantly outperforms Spann3R demonstrating its scalability to large scenes.

	Reg.	RRA@5	RTA@5	RRA@15	RTA@15	ATE
Indp.	1.0	77.21	83.25	94.13	93.21	0.0175
Merged	0.87	74.65	79.69	94.83	93.89	0.0165

Table 2. Comparison of pose accuracy when processing each image collections vs. all image collections merged into one.

D. Handling disconnected image collections.

In many real world scenarios image collections might either contain invalid images, e.g., images that are not visually overlapping with the contents depcited in most of the other images, or the image collection might itself contain subsets of images showing different scenes.

To investigate the robustness of Light3R-SfM in these cases we create such a scenario by merging all *eight* 25-view scenes from the Tanks&Temples intermediate split into a single image collection. We then identify disconnected sub-reconstructions by filtering edges with an average pointmap confidence below 3. For each scene, we retain only the largest sub-reconstruction, treating all other images as unregistered.

As shown in Tab. 2, our method successfully registers 87% of all images across *eight* scenes while maintaining similar accuracy to processing each scene independently. This demonstrates that (i) our graph construction ensures locally consistent connections and (ii) our learned confidence maps effectively reject artificial connections. We agree that adaptively constructing a new view-graph is a promising direction for further improvement.

E. Drift analysis.

We qualitatively compare ground truth and predicted camera trajectories on the held-out ”Gascola” scene from TartanAir (as test set ground truth poses are unavailable). Running our method on shorter sequences, like *Easy/P001*, yields drift-free trajectories. For the sequence *Hard/P009* with loop closures, we use a five-frame sliding window for all methods to mitigate ambiguity from extreme visual similarity. As shown in Fig. 4, Light3R-SfM exhibits some drift, however, it remains more robust than other methods—GLOMAP recovers only 464 of 764 views, while MASt3R-SfM and Spann3R fail to recover any trajectory.

F. Additional Ablation Studies

In addition to the ablation studies performed in the main paper, we consider more detailed ablations for hyper-parameters specific to our contributions. To save compute,

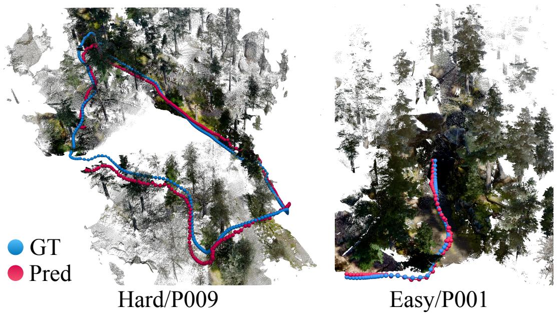

Figure 4. Predicted camera trajectories for two scenes of the TartanAir dataset.

	RRA@5 ↑	RTA@5 ↑	ATE ↓
L			
2	33.1	35.5	0.033
4	**35.7**	**36.9**	**0.032**
8	35.3	**36.9**	**0.032**

Table 3. **Impact of number of latent alignment layers L.**

we train the models for these ablation studies on lower resolution images, *i.e.*, 224×224, versus a maximum resolution of 512×384 for the results reported in the main paper. For these experiments, we report results on the 100-view subset of Tanks&Temples unless otherwise stated.

Global alignment layers. For results reported in the main paper, we always consider $L = 4$ latent global alignment layers. Here we ablate this choice by considering $L \in \{2, 4, 8\}$. In Tab. 3, we report pose accuracy metrics for the different settings of L. Using 4 latent alignment layers significantly improves results compared to 2, but doubling the number shows diminishing returns, leading us to select $L = 4$ as a trade-off between memory usage/runtime and pose accuracy.

Weight of global supervision. For the loss supervising the globally aligned pointmaps, accumulated from the pairwise reconstructions, we consider $\lambda = 0.1$ as the default. Here, we experiment with other choices of λ.

In Tab. 4, we report pose accuracy metrics for choices $\lambda \in \{0.01, 0.1, 1.0\}$. We find that increasing the loss weight from 0.01 to 0.1 improves pose estimation, however, the higher setting of $\lambda = 1.0$ decreases performance. We explain this behavior with the fact that the global loss produces more noisy supervision compared to the pairwise loss: if a pairwise reconstruction is incorrect it will potentially affects global pointmaps of other views due to global accumulation. Thus, it is beneficial when the pairwise supervision is the main driver of the optimization of model parameters where the global supervision acts as a contributing signal with relative lower weight.

	RRA@5 ↑	RTA@5 ↑	ATE ↓
λ			
0.01	30.6	33.3	0.034
0.1	**35.7**	**36.9**	0.032
1	34.5	36.8	**0.031**

Table 4. **Impact of weight of global supervision λ.**

	RRA@5 ↑	RTA@5 ↑	ATE ↓
Backbone init.			
Scratch	0.7	0.1	0.057
DUSt3R	**35.7**	36.9	**0.032**
MASt3R	34.6	**38.6**	**0.032**

Table 5. **Impact of backbone initialization.**

	RRA@5 ↑	RTA@5 ↑	ATE ↓
N			
3	37.3	38.1	0.033
5	38.0	39.4	**0.030**
8	39.0	39.5	**0.030**
10	**39.2**	**40.7**	0.031

Table 6. **Impact of training graph size N.** We report results averaged over Tanks&Temples scenes with all frames.

Number of images in training graph. All results in the main paper are achieved with models optimized with training graphs of $N = 8$ images. In Tab. 6, we report results for $N \in \{3, 5, 8, 10\}$. To achieve a fair comparison we increase the batch size for smaller settings of N such that the total number of images per batch and seen over the course of training remains the same. Overall, we find a small but consistent improvement for larger training graphs. We explain this consistent improvement by the number of relative constraints in the training graph increasing as the size of the graph increases. With global supervision enforcing consistency of these pairwise constraints the latent alignment layers experience additional supervision leading to better downstream performance. While we achieve better performance on $N = 10$, we use $N = 8$ for the higher resolution model in the main paper since larger training graphs exceed the GPU memory capacity.

Model initialization. In Tab. 5, we report results with different pre-trained weights for the pairwise pointmap regressor used within our method. We find that initializing with either MASt3R [8] or the DUSt3R [20] backbone leads to comparable results. If we train the pairwise regressor from scratch, jointly with the other components, we observe that the model performs poorly. This highlights the significance of building on top of geometric foundation models as components for our approach.

G. Additional Visualizations

Reconstruction examples. We provide visualizations of reconstructions obtained using Light3R-SfM. In Fig. 5, we show reconstruction of diverse Tanks&Temples scenes, including indoor, object-centric, and large scale reconstructions of landmarks. Further, we provide qualitative results on the challenging ETH3D [13] scenes in Fig. 6.

Qualitative comparisons on Waymo sequences. We provide additional qualitative comparison of 3D reconstructions from the Waymo Open Dataset [15] obtained by MASt3R-SfM, Spann3R, and Light3R-SfM. As shown in Fig. 7, Spann3R fails to reconstruct the camera poses as well as the scene structure when the trajectory is longer, while MASt3R-SfM fails to recover the boundary and further away background regions, leading to noisy and coarse reconstruction. In contrast, our method is able to recover accurate camera poses as well as capture fine details in the scene, *e.g.*, cars and buildings along the street.

Failure cases. Finally, we provide visualizations of typical failure cases in Fig. 8. We observed that retrieval failures can result in multiple sub-reconstructions which are aligned within themselves but globally inconsistent. Further, small errors in the pairwise estimations result in misalignment in the global reconstruction.

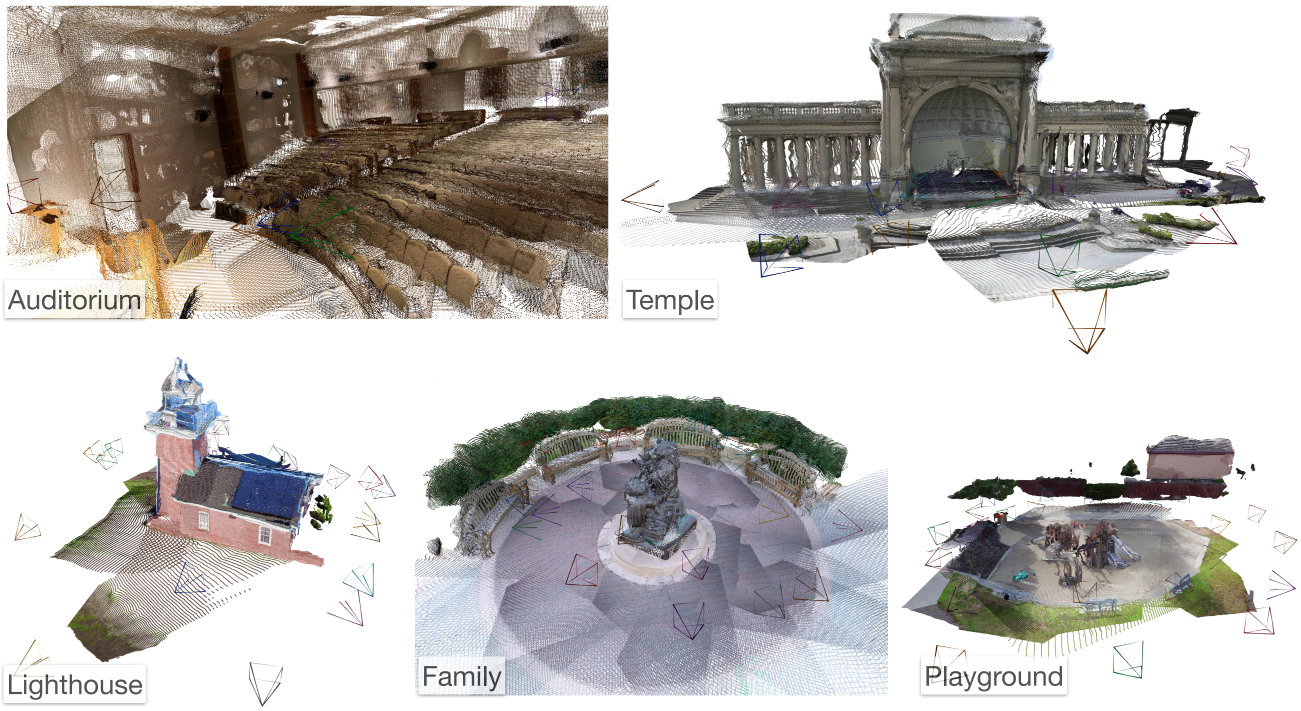

Figure 5. **Qualitative examples of reconstruction of Tanks & Temples scenes.**

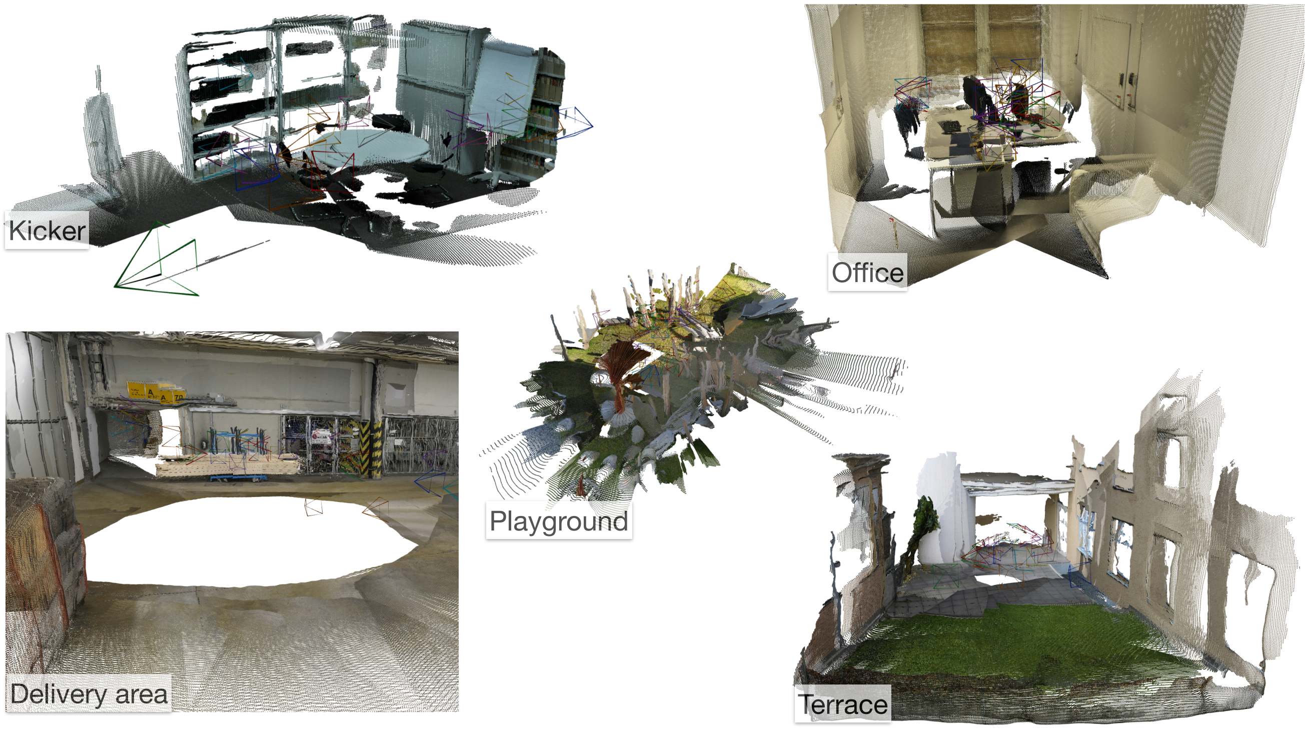

Figure 6. **Qualitative examples of reconstruction of ETH3D scenes.**

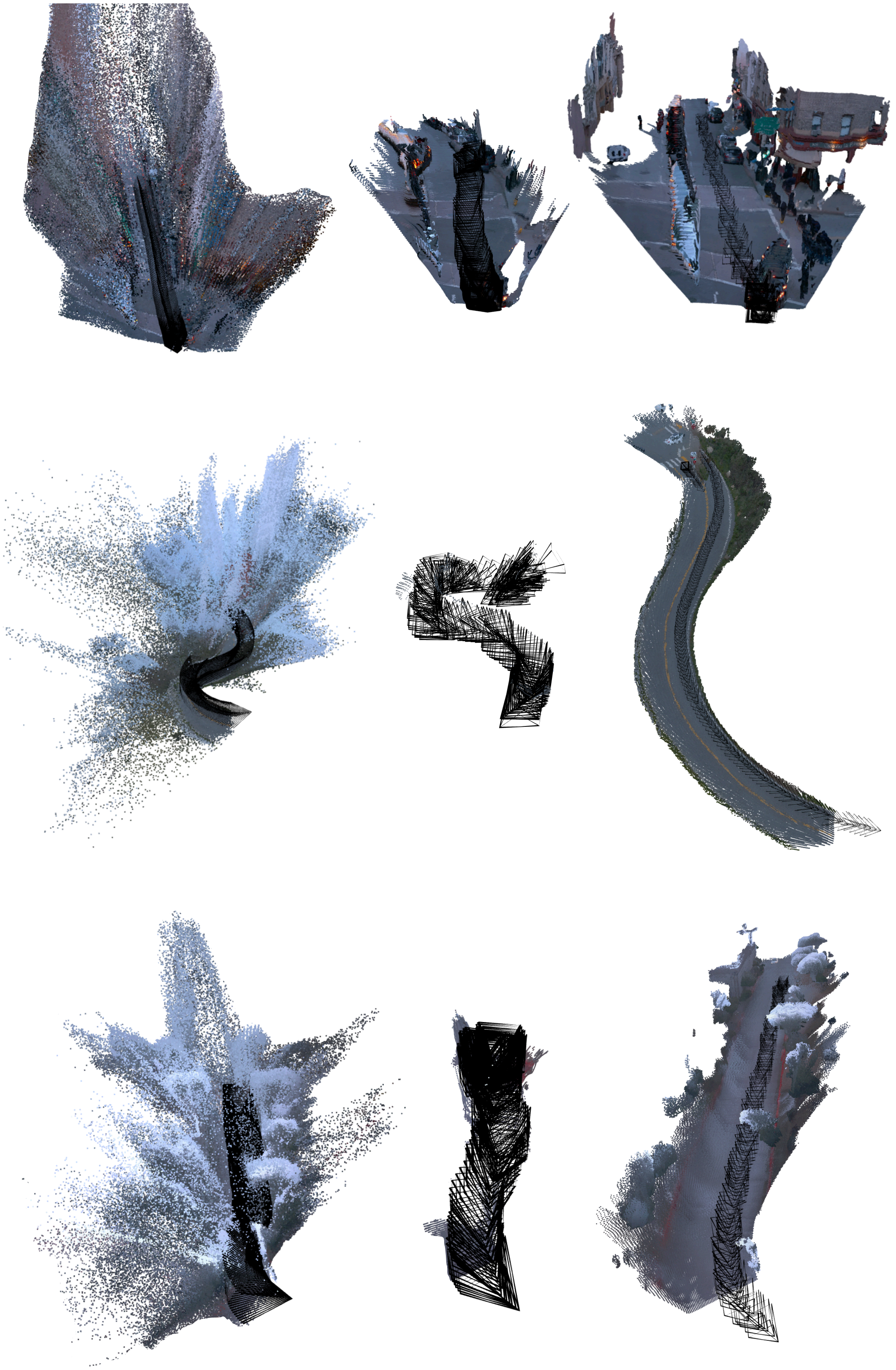

Figure 7. **More comparisons on Waymo.** Comparing from left-to-right: MASt3R-SfM, Spann3R, Light3R-SfM.

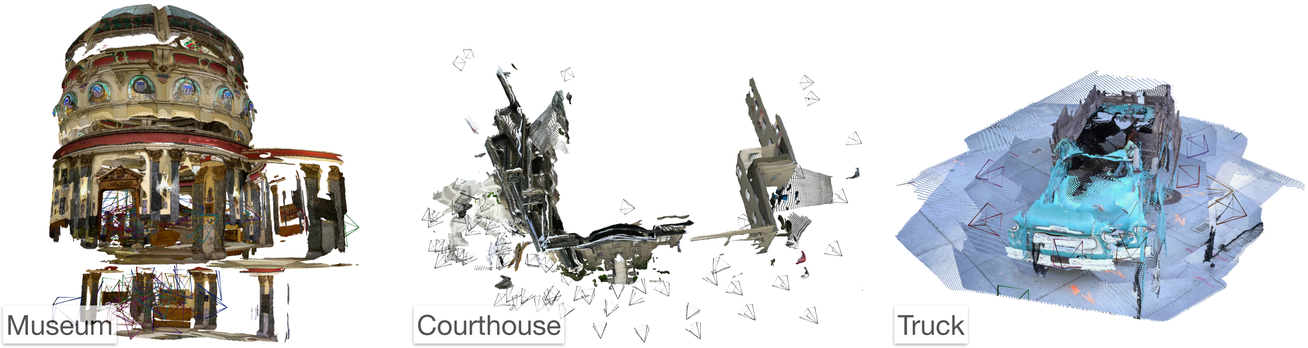

Figure 8. **Failure cases on the Tanks & Temples dataset.**

#images	Group	Scene	ACE0 [2]	MASt3R-SfM [4]	VGGSfM [19]	GLOMAP [11]	Spann3r [18]	**Light3R-SfM**
25	Advanced	Auditorium	1113.5	288.1	90.7	10.4	10.3	5.8
		Ballroom	1141.6	294.3	167.3	25.4	7.6	4.3
		Courtroom	1534.4	290.3	125.2	10.9	7.1	4.3
		Museum	917.4	287.0	155.2	15.5	8.7	4.4
		Palace	-	286.3	219.7	7.3	7.3	3.9
		Temple	-	285.0	241.2	9.4	7.7	4.1
	Intermediate	Family	1334.7	285.4	83.5	20.1	10.9	5.6
		Francis	970.6	277.8	80.1	13.0	7.5	3.9
		Horse	980.2	287.1	66.9	29.3	11.8	5.5
		Lighthouse	881.5	282.0	110.7	7.6	7.3	3.7
		M60	826.8	268.0	191.7	22.3	7.0	3.8
		Panther	831.1	268.5	140.5	14.4	7.0	4.0
		Playground	866.5	291.1	97.6	8.0	8.3	4.4
		Train	-	290.5	114.2	19.8	7.8	4.4
	Train	Barn	986.6	273.6	140.9	10.4	8.8	4.4
		Caterpillar	1229.5	285.4	79.3	15.1	7.2	4.4
		Church	1088.3	268.4	111.4	24.0	9.9	4.4
		Courthouse	-	293.0	328.4	18.0	8.8	3.7
		Ignatius	931.6	262.5	75.7	16.9	6.9	4.3
		Meetingroom	1165.0	280.7	127.7	12.2	9.1	4.1
		Truck	926.3	301.8	102.3	27.8	8.0	4.6
50	Advanced	Auditorium	1497.2	532.4	282.0	32.5	20.5	10.5
		Ballroom	1848.6	529.1	272.5	110.9	14.7	9.0
		Courtroom	1480.7	505.6	415.5	32.6	16.3	8.8
		Museum	992.5	468.8	670.5	48.4	17.8	8.1
		Palace	-	486.1	601.1	21.7	16.3	7.6
		Temple	-	481.7	499.9	25.0	16.3	8.5
	Intermediate	Family	2140.4	491.7	143.8	81.2	21.8	11.4
		Francis	1226.7	498.8	120.0	47.2	14.5	7.3
		Horse	2604.0	497.3	184.6	68.5	22.6	10.8
		Lighthouse	1324.7	476.6	270.6	40.8	14.7	7.2
		M60	1304.3	457.4	267.9	34.6	14.3	7.2
		Panther	2072.7	456.4	178.4	52.0	13.9	7.2
		Playground	1105.8	499.8	202.3	25.6	17.8	8.4
		Train	1097.9	526.7	217.0	41.3	14.7	8.4
	Train	Barn	973.3	522.5	340.2	30.8	16.5	8.0
		Caterpillar	1110.3	530.9	151.7	37.4	14.4	8.2
		Church	1551.0	505.0	413.9	38.5	19.1	8.4
		Courthouse	1014.6	508.3	305.9	41.3	17.2	8.0
		Ignatius	2724.5	485.2	144.7	47.2	14.4	8.3
		Meetingroom	1451.1	566.7	252.2	72.8	19.2	8.1
		Truck	1549.9	535.1	183.0	67.8	13.4	8.7
100	Advanced	Auditorium	4885.4	931.7	740.7	85.2	37.9	19.2
		Ballroom	1987.4	909.4	732.2	448.7	26.5	17.5
		Courtroom	4942.5	931.9	850.2	127.7	28.9	17.0
		Museum	5031.7	848.5	768.0	172.2	34.6	16.5
		Palace	1670.7	885.1	1934.4	87.4	32.2	15.7
		Temple	1126.2	831.4	1169.3	97.8	31.9	16.5
	Intermediate	Family	2275.3	890.7	391.7	236.3	42.8	22.3
		Francis	4689.7	909.8	365.1	147.3	26.5	14.8
		Horse	3882.0	909.5	351.9	189.2	43.5	21.1
		Lighthouse	2590.9	845.2	512.1	128.9	28.5	14.0
		M60	1800.5	813.5	522.6	127.2	25.9	14.5
		Panther	1659.3	798.0	488.4	173.9	26.3	14.5
		Playground	1303.4	888.5	456.9	100.8	33.0	17.0
		Train	4441.7	930.8	789.5	144.3	29.5	16.4
	Train	Barn	7502.6	811.7	605.9	98.0	30.7	16.6
		Caterpillar	5145.0	857.4	373.1	120.9	25.8	16.8
		Church	3242.8	821.2	782.0	201.9	34.5	16.6
		Courthouse	990.9	838.3	1043.8	127.5	32.0	15.2
		Ignatius	2378.6	758.4	379.3	156.1	25.4	16.3
		Meetingroom	6338.4	874.5	526.6	203.5	31.8	16.2
		Truck	3367.2	805.5	495.9	234.2	25.3	17.2
200	Advanced	Auditorium	5388.1	1748.6	2185.9	349.0	71.1	37.1
		Ballroom	3782.7	1680.6	1779.0	1466.7	53.1	35.3
		Courtroom	3245.0	1758.6	1886.8	453.8	58.1	34.6
		Museum	3952.1	1661.4	2162.8	634.0	68.7	33.2
		Palace	3267.1	1747.5	3910.8	324.6	64.0	31.3
		Temple	1480.1	1659.4	2221.5	309.2	59.0	33.3
	Intermediate	Family	2349.0	1598.7	679.2	667.0	83.7	44.3
		Francis	3522.3	1654.0	846.6	462.9	55.8	29.6
		Horse	4176.3	1610.1	624.4	558.8	88.9	41.9
		Lighthouse	9072.2	1573.2	1411.4	455.9	54.5	28.0
		M60	5080.5	1481.3	1198.5	453.5	49.9	28.8
		Panther	3837.3	1461.8	1134.0	560.2	49.8	28.7
		Playground	8317.4	1578.9	1010.1	360.0	61.4	33.1
		Train	4022.3	1585.0	1638.5	503.0	54.3	32.9
	Train	Barn	6885.8	1504.8	-	349.2	56.6	33.5
		Caterpillar	5360.8	1610.9	1066.4	386.8	49.0	33.9
		Church	4418.7	1577.1	-	522.5	69.7	34.3
		Courthouse	8604.2	1598.8	-	358.5	59.4	30.8
		Ignatius	2565.2	1521.1	873.9	533.0	51.9	31.5
		Meetingroom	4025.3	1645.4	-	697.6	59.6	32.0
		Truck	3340.0	1531.7	1067.0	865.3	50.2	34.5
full	Advanced	Auditorium	4713.9	2349.4	2739.6	544.8	76.8	46.5
		Ballroom	4462.5	2639.0	-	3407.7	83.6	55.7
		Courtroom	3907.0	2556.5	-	975.6	80.5	50.4
		Museum	5030.7	2422.9	-	1168.9	95.2	49.5
		Palace	3856.3	3519.4	-	1537.4	132.5	66.6
		Temple	6103.9	2187.3	-	758.2	85.9	47.6
	Intermediate	Family	3655.3	3696.1	-	3396.7	207.5	110.1
		Francis	4380.5	2369.8	-	961.9	80.0	45.2
		Horse	4589.4	3641.7	-	2507.3	223.4	100.4
		Lighthouse	8594.3	2244.1	-	940.7	84.8	41.4
		M60	5796.9	2174.1	-	1094.1	80.2	44.4
		Panther	4102.6	2174.4	-	1317.5	77.8	44.4
		Playground	7335.7	2316.1	-	894.1	104.0	49.3
		Train	7334.8	2487.8	-	1099.4	84.3	48.2
	Train	Barn	8072.1	2933.4	-	1558.5	123.1	66.1
		Caterpillar	5909.6	2989.8	-	1182.6	93.8	62.9
		Church	5491.6	3590.2	-	2988.4	175.1	84.4
		Courthouse	11817.4	-	-	10656.7	318.5	177.4
		Ignatius	2407.4	-	-	844.2	67.1	40.6
		Meetingroom	4440.0	-	-	2422.2	103.9	58.7
		Truck	3488.5	-	1528.8	1274.3	63.3	41.5

Table 7. **Per-scene reconstruction runtimes on Tanks&Temples.** All runtimes are reported in seconds.

References

[1] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf. Parametric correspondence and chamfer matching: Two new techniques for image matching. In *Proceedings of the 5th International Joint Conference on Artificial Intelligence - Volume 2*, pages 659–663. 2

[2] 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. In *European Conference on Computer Vision*, 2024. 9

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*, 2020. 1

[4] Bardienus Duisterhof, Lojze Zust, Philippe Weinzaepfel, Vincent Leroy, Yohann Cabon, and Jerome Revaud. Mast3r-sfm: a fully-integrated solution for unconstrained structure-from-motion. *arXiv preprint arXiv:2409.19152*, 2024. 9

[5] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, (6):381–395, 1981. 1

[6] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2

[7] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36(4), 2017. 1

[8] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. *European Conference on Computer Vision*, 2024. 2, 4

[9] Zhengqi Li and Noah Snavely. MegaDepth: Learning Single-View Depth Prediction From Internet Photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2041–2050, 2018. 1

[10] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization, 2019. 1

[11] Linfei Pan, Dániel Baráth, Marc Pollefeys, and Johannes L Schönberger. Global structure-from-motion revisited. In *European Conference on Computer Vision (ECCV)*, 2024. 9

[12] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common Objects in 3D: Large-Scale Learning and Evaluation of Real-Life 3D Category Reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10901–10911, 2021. 1

[13] Thomas Schops, Johannes L. Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A Multi-View Stereo Benchmark With High-Resolution Images and Multi-Camera Videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3260–3269, 2017. 5

[14] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2930–2937. 3

[15] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2446–2454, 2020. 1, 2, 3, 5

[16] George Terzakis and Manolis Lourakis. A Consistently Fast and Globally Optimal Solution to the Perspective-n-Point Problem. In *Computer Vision – ECCV 2020*, pages 478–494, Cham, 2020. 1

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, 2017. 1

[18] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. *arXiv preprint arXiv:2408.16061*, 2024. 2, 9

[19] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. Vggsfm: Visual geometry grounded deep structure from motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21686–21697, 2024. 2, 9

[20] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024. 1, 4

[21] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. TartanAir: A Dataset to Push the Limits of Visual SLAM. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4909–4916, 2020. 1

[22] Philippe Weinzaepfel, Thomas Lucas, Vincent Leroy, Yohann Cabon, Vaibhav Arora, Romain Brégier, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jerome Revaud. CroCo v2: Improved Cross-view Completion Pretraining for Stereo Matching and Optical Flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17969–17980, 2023. 1

[23] E. Weiszfeld. Sur le point pour lequel la Somme des distances de n points donnés est minimum. *Tohoku Mathematical Journal, First Series*, pages 355–386, 1937. 1