VGGT: Visual Geometry Grounded Transformer

Supplementary Material

Appendix

In the Appendix, we provide the following:

- formal definitions of key terms in Appendix A.
- comprehensive implementation details, including architecture, training losses, and hyperparameters in Appendix B.
- additional experiments in Appendix C.
- more qualitative examples, including single-view reconstruction, in Appendix D.
- an expanded review of related works in Appendix E.
- additional discussions in Appendix F.

A. Formal Definitions

In this section, we provide additional formal definitions that further ground the method section.

The camera extrinsics are defined in relation to the *world reference frame*, which we take to be the coordinate system of the first camera. We thus introduce two functions. The first function $\gamma(\mathbf{g}, \mathbf{p}) = \mathbf{p}'$ applies the rigid transformation encoded by $\mathbf{g}$ to a point $\mathbf{p}$ in the world reference frame to obtain the corresponding point $\mathbf{p}'$ in the camera reference frame. The second function $\pi(\mathbf{g}, \mathbf{p}) = \mathbf{y}$ further applies perspective projection, mapping the 3D point $\mathbf{p}$ to a 2D image point $\mathbf{y}$. We also denote the depth of the point as observed from the camera $\mathbf{g}$ by $\pi^{\mathrm{D}}(\mathbf{g}, \mathbf{p}) = d \in \mathbb{R}^+$.

We model the scene as a collection of regular surfaces $S_i \subset \mathbb{R}^3$. We make this a function of the i-th input image as the scene can change over time [77]. The depth at pixel location $\mathbf{y} \in \mathcal{I}(I_i)$ is defined as the minimum depth of any 3D point $\mathbf{p}$ in the scene that projects to $\mathbf{y}$, *i.e.*, $D_i(\mathbf{y}) = \min\{\pi^D(\mathbf{g}_i, \mathbf{p}) : \mathbf{p} \in S_i \ \wedge \ \pi(\mathbf{g}_i, \mathbf{p}) = \mathbf{y}\}$. The point at pixel location $\mathbf{y}$ is then given by $P_i(\mathbf{y}) = \gamma(\mathbf{g}, \mathbf{p})$, where $\mathbf{p} \in S_i$ is the 3D point that minimizes the expression above, *i.e.*, $\mathbf{p} \in S_i \ \wedge \ \pi(\mathbf{g}_i, \mathbf{p}) = \mathbf{y} \ \wedge \ \pi^D(\mathbf{g}_i, \mathbf{p}) = D_i(\mathbf{y})$.

B. Implementation Details

Architecture. As mentioned in the main paper, VGGT consists of 24 attention blocks, each block equipped with one frame-wise self-attention layer and one global self-attention layer. Following the ViT-L model used in DINOv2 [37], each attention layer is configured with a feature dimension of 1024 and employs 16 heads. We use the official implementation of the attention layer from PyTorch, *i.e.*, *torch.nn.MultiheadAttention*, with flash attention enabled. To stabilize training, we also use QKNorm [23] and LayerScale [61] for each attention layer. The value of LayerScale is initialized with 0.01. For image tokenization, we use DINOv2 [37] and add positional embedding. As in [74], we feed the tokens from the 4-th, 11-th, 17-th, and 23-rd block into DPT [43] for upsampling.

Training Losses. We train the VGGT model f end-to-end using a multi-task loss:

$$\mathcal{L} = \mathcal{L}_{\text{camera}} + \mathcal{L}_{\text{depth}} + \mathcal{L}_{\text{pmap}} + \lambda \mathcal{L}_{\text{track}}. \tag{1}$$

We found that the camera ($\mathcal{L}_{\text{camera}}$), depth ($\mathcal{L}_{\text{depth}}$), and point-map ($\mathcal{L}_{\text{pmap}}$) losses have similar ranges and do not need to be weighted against each other. The tracking loss $\mathcal{L}_{\text{track}}$ is down-weighted with a factor of $\lambda = 0.05$. We describe each loss term in turn.

The camera loss $\mathcal{L}_{\text{camera}}$ supervises the cameras $\hat{\mathbf{g}}$: $\mathcal{L}_{\text{camera}} = \sum_{i=1}^{N} \|\hat{\mathbf{g}}_i - \mathbf{g}_i\|_\epsilon$, comparing the predicted cameras $\hat{\mathbf{g}}_i$ with the ground truth $\mathbf{g}_i$ using the Huber loss $|\cdot|_\epsilon$.

The depth loss $\mathcal{L}_{\text{depth}}$ follows DUSt3R [67] and implements the aleatoric-uncertainty loss [28, 36] weighing the discrepancy between the predicted depth $\hat{D}_i$ and the ground-truth depth D_i with the predicted uncertainty map $\hat{\Sigma}_i^D$. Differently from DUSt3R, we also apply a gradient-based term, which is widely used in monocular depth estimation. Hence, the depth loss is $\mathcal{L}_{\text{depth}} = \sum_{i=1}^{N} \|\Sigma_i^D \odot (\hat{D}_i - D_i)\| + \|\Sigma_i^D \odot (\nabla \hat{D}_i - \nabla D_i)\| - \alpha \log \Sigma_i^D$, where $\odot$ is the channel-broadcast element-wise product. The point map loss is defined analogously but with the point-map uncertainty Σ_i^P: $\mathcal{L}_{\text{pmap}} = \sum_{i=1}^{N} \|\Sigma_i^P \odot (\hat{P}_i - P_i)\| + \|\Sigma_i^P \odot (\nabla \hat{P}_i - \nabla P_i)\| - \alpha \log \Sigma_i^P$.

Finally, the tracking loss is given by $\mathcal{L}_{\text{track}} = \sum_{j=1}^{M} \sum_{i=1}^{N} \|\mathbf{y}_{j,i} - \hat{\mathbf{y}}_{j,i}\|$. Here, the outer sum runs over all ground-truth query points $\mathbf{y}_j$ in the query image I_q, $\mathbf{y}_{j,i}$ is $\mathbf{y}_j$'s ground-truth correspondence in image I_i, and $\hat{\mathbf{y}}_{j,i}$ is the corresponding prediction obtained by the application $\mathcal{T}((\mathbf{y}_j)_{j=1}^M, (T_i)_{i=1}^N)$ of the tracking module. Additionally, following CoTracker2 [27], we apply a visibility loss (binary cross-entropy) to estimate whether a point is visible in a given frame.

Ground Truth Coordinate Normalization. If we scale a scene or change its global reference frame, the images of the scene are not affected at all, meaning that any such variant is a legitimate result of 3D reconstruction. We remove this ambiguity by normalizing the data, thus making a canonical choice and task the transformer to output this particular variant. We follow [67] and, first, express all quantities in the coordinate frame of the first camera $\mathbf{g}_1$. Then, we compute the average Euclidean distance of all 3D points in the point map P to the origin and use this scale to normalize the camera translations $\mathbf{t}$, the point map P, and the depth map

D. Importantly, unlike [67], we do *not* apply such normalization to the predictions output by the transformer; instead, we force it to learn the normalization we choose from the training data.

Implementation Details. By default, we employ $L = 24$ layers of global and frame-wise attention, respectively. The model consists of approximately 1.2 billion parameters in total. We train the model by optimizing the training loss (1) with the AdamW optimizer for 160K iterations. We use a cosine learning rate scheduler with a peak learning rate of 0.0002 and a warmup of 8K iterations. For every batch, we randomly sample 2–24 frames from a random training scene. The input frames, depth maps, and point maps are resized to a maximum dimension of 518 pixels. The aspect ratio is randomized between 0.33 and 1.0. We also randomly apply color jittering, Gaussian blur, and grayscale augmentation to the frames. The training runs on 64 A100 GPUs over nine days. We employ gradient norm clipping with a threshold of 1.0 to ensure training stability. We leverage bfloat16 precision and gradient checkpointing to improve GPU memory and computational efficiency.

In more details, to form a training batch, we first choose a random training dataset (each dataset has a different yet approximately similar weight, as in [67]), and from the dataset, we then sample a random scene (uniformly). During the training phase, we select between 2 and 24 frames per scene while maintaining the constant total of 48 frames within each batch. For training, we use the respective training sets of each dataset. We exclude training sequences containing fewer than 24 frames. RGB frames, depth maps, and point maps are first isotropically resized, so the longer size has 518 pixels. Then, we crop the shorter dimension (around the principal point) to a size between 168 and 518 pixels while remaining a multiple of the 14-pixel patch size. It is worth mentioning that we apply aggressive color augmentation independently across each frame within the same scene, enhancing the model's robustness to varying lighting conditions. We build ground truth tracks following [18, 54, 66], which unprojects depth maps to 3D, reprojects points to target frames, and retains correspondences where reprojected depths match target depth maps. Frames with low similarity to the query frame are excluded during batch sampling. In rare cases with no valid correspondences, the tracking loss is omitted.

Training Data. The model was trained using a large and diverse collection of datasets, including: Co3Dv2 [44], BlendMVS [75], DL3DV [33], MegaDepth [30], Kubric [20], WildRGB [70], ScanNet [13], HyperSim [45], Mapillary [34], Habitat [56], Replica [53], MVS-Synth [24], PointOdyssey [80], Virtual KITTI [6], Aria Synthetic Environments [40], Aria Digital Twin [40], and a synthetic dataset of artist-created assets similar to Objaverse [14]. These datasets span various domains, including indoor and outdoor environments, and encompass synthetic and real-world scenarios. The 3D annotations for these datasets are derived from multiple sources, such as direct sensor capture, synthetic engines, or SfM techniques [48]. The combination of our datasets is broadly comparable to those of MASt3R [17] in size and diversity.

C. Additional Experiments

Camera Pose Estimation on IMC We also evaluate using the Image Matching Challenge (IMC) [26], a camera pose estimation benchmark focusing on phototourism data. Until recently, the benchmark was dominated by classical incremental SfM methods [47].

Baselines. We evaluate two flavors of our model: VGGT and VGGT + BA. VGGT directly outputs camera pose estimates, while VGGT + BA refines the estimates using an additional Bundle Adjustment stage. We compare to the classical incremental SfM methods such as [32, 47] and to recently-proposed deep methods. Specifically, recently VGGSfM [66] provided the first end-to-end trained deep method that outperformed incremental SfM on the challenging phototourism datasets.

Besides VGGSfM, we additionally compare to recently popularized DUSt3R [67] and MASt3R [29]. It is important to note that DUSt3R and MASt3R utilized a substantial portion of the MegaDepth dataset for training, only excluding scenes 0015 and 0022. The MegaDepth scenes employed in their training have some overlap with the IMC benchmark, although the images are not identical; the same scenes are present in both datasets. For instance, the MegaDepth scene 0024 corresponds to the British Museum, while the British Museum is also a scene in the IMC benchmark. For an apples-to-apples comparison, we adopt the same training split as DUSt3R and MASt3R. In the main paper, to ensure a fair comparison on ScanNet-1500, we exclude the corresponding ScanNet scenes from our training.

Results. Table A contains the results of our evaluation. Although phototourism data is the traditional focus of SfM methods, our VGGT's feed-forward performance is on par with the state-of-the-art VGGSfMv2 with AUC@10 of 71.26 versus 76.82, while being significantly faster (0.2 vs. 10 seconds per scene). Remarkably, VGGT outperforms both MASt3R [29] and DUSt3R [67] significantly across all accuracy thresholds while being much faster. This is because MASt3R's and DUSt3R's feed-forward predictions can only process pairs of frames and, hence, require a costly global alignment step. Additionally, with bundle adjustment, VGGT + BA further improves drastically, achieving state-of-the-art performance on IMC, raising AUC@10 from 71.26 to 84.91, and raising AUC@3 from 39.23 to 66.37. Note that our model directly predicts 3D points,

Figure A. **Single-view Reconstruction by Point Map Estimation.** Unlike DUSt3R, which requires duplicating an image into a pair, our model can predict the point map from a single input image. It demonstrates strong generalization to unseen real-world images.

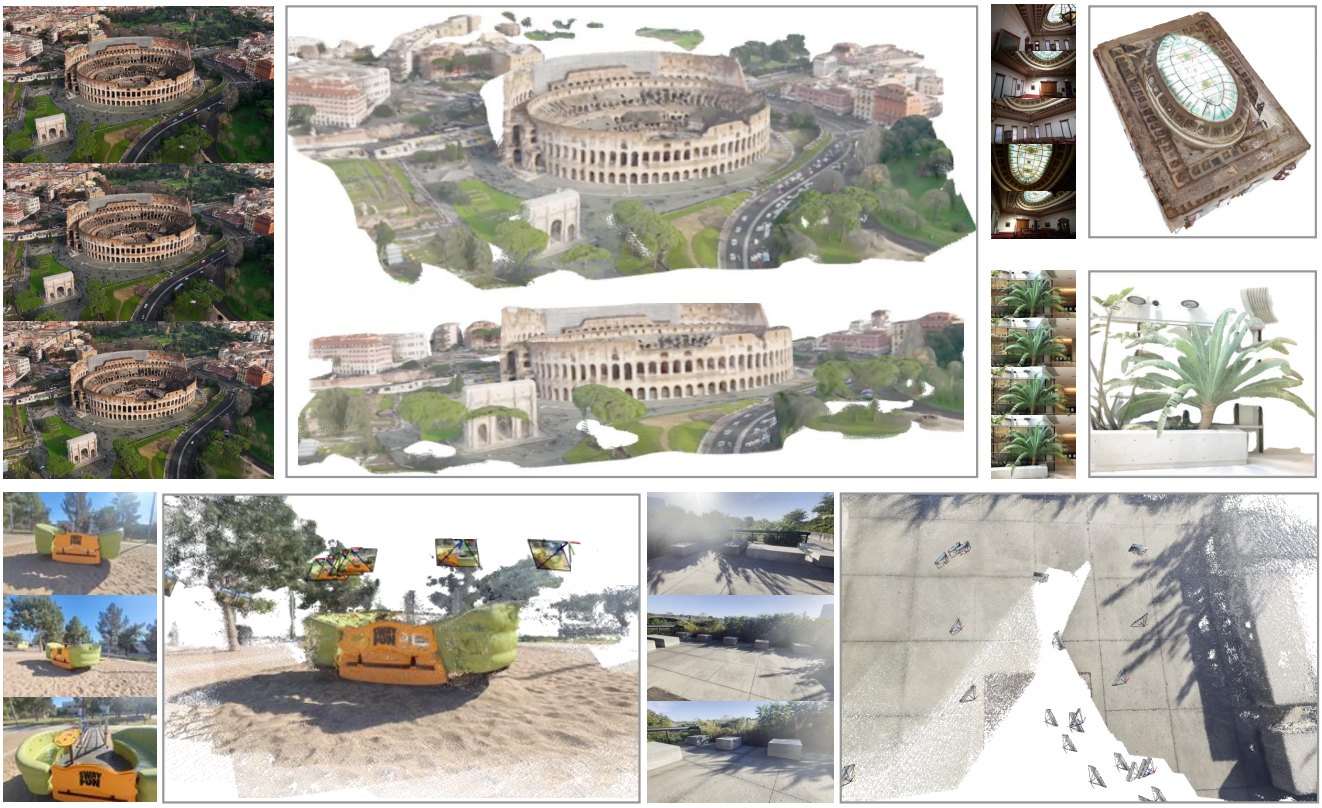

Figure B. **Additional Visualizations of Point Map Estimation.** Camera frustums illustrate the estimated camera poses. Explore our interactive demo for better visualization quality.

which can serve as the initialization for BA. This eliminates the need for triangulation and iterative refinement of BA as in [66]. As a result, VGGT + BA is much faster than [66].

D. Qualitative Examples

We additionally present qualitative examples in Fig. B, along with single-view reconstruction results in Fig. A.

E. Related Work

In this section, we discuss additional related works.

Vision Transformers. The Transformer architecture was initially proposed for language processing tasks [5, 15, 63]. It was later introduced to the computer vision community by ViT [16], sparking widespread adoption. Vision Transformers and their variants have since become dominant in the design of architectures for various computer vision

Method	Test-time Opt.	AUC@3°	AUC@5°	AUC@10°	Runtime
COLMAP (SIFT+NN) [47]	✓	23.58	32.66	44.79	>10s
PixSfM (SIFT + NN) [32]	✓	25.54	34.80	46.73	>20s
PixSfM (LoFTR) [32]	✓	44.06	56.16	69.61	>20s
PixSfM (SP + SG) [32]	✓	45.19	57.22	70.47	>20s
DFSfM (LoFTR) [22]	✓	46.55	58.74	72.19	>10s
DUSt3R [67]	✓	13.46	21.24	35.62	∼ 7s
MASt3R [29]	✓	30.25	46.79	57.42	∼ 9s
VGGSfM [66]	✓	45.23	58.89	73.92	∼ 6s
VGGSfMv2 [66]	✓	59.32	67.78	76.82	∼ 10s
VGGT (ours)	✗	39.23	52.74	71.26	**0.2s**
VGGT + BA (ours)	✓	**66.37**	**75.16**	**84.91**	1.8s

Table A. **Camera Pose Estimation on IMC [26].** Our method achieves state-of-the-art performance on the challenging phototropism data, outperforming VGGSfMv2 [66] which ranked first on the latest CVPR'24 IMC Challenge in camera pose (rotation and translation) estimation.

tasks [3, 8, 41, 71], thanks to their simplicity, high capacity, flexibility, and ability to capture long-range dependencies.

DeiT [60] demonstrated that Vision Transformers can be effectively trained on datasets like ImageNet using strong data augmentation strategies. DINO [7] revealed intriguing properties of features learned by Vision Transformers in a self-supervised manner. CaiT [61] introduced layer scaling to address the challenges of training deeper Vision Transformers, effectively mitigating gradient-related issues. Further, techniques such as QKNorm [23, 76] have been proposed to stabilize the training process. Additionally, [72] also explores the dynamics between frame-wise and global attention modules in object tracking, though using cross-attention.

Camera Pose Estimation. Estimating camera poses from multi-view images is a crucial problem in 3D computer vision. Over the last decades, Structure from Motion (SfM) has emerged as the dominant approach [21], whether incremental [1, 19, 47, 52, 69] or global [2, 9–12, 25, 35, 38, 39, 46, 55]. Recently, a set of methods treat camera pose estimation as a regression problem [31, 50, 57–59, 62, 64, 65, 68, 78, 79, 81], which show promising results under the sparse-view setting. Ace-Zero [4] further proposes to regress 3D scene coordinates and FlowMap [51] focuses on depth maps, as intermediates for camera prediction. Instead, VGGSfM [66] simplifies the classical SfM pipeline to a differentiable framework, demonstrating exceptional performance, particularly with phototourism datasets. At the same time, DUSt3R [29, 67] introduces an approach to learn pixel-aligned point map, and hence camera poses can be recovered by simple alignment. This paradigm shift has garnered considerable interest as the point map, an over-parameterized representation, offers seamless integration with various downstream applications, such as 3D Gaussian splatting.

Input Frames	1	2	4	8	10	20	50	100	200
Time (s)	0.04	0.05	0.07	0.11	0.14	0.31	1.04	3.12	8.75
Mem. (GB)	1.88	2.07	2.45	3.23	3.63	5.58	11.41	21.15	40.63

Table B. **Runtime and peak GPU memory usage across different numbers of input frames.** Runtime is measured in seconds, and GPU memory usage is reported in gigabytes.

F. Discussions

Limitations. While our method exhibits strong generalization to diverse in-the-wild scenes, several limitations remain. First, the current model does not support fisheye or panoramic images. Additionally, reconstruction performance drops under conditions involving extreme input rotations. Moreover, although our model handles scenes with minor non-rigid motions, it fails in scenarios involving substantial non-rigid deformation.

However, an important advantage of our approach is its flexibility and ease of adaptation. Addressing these limitations can be straightforwardly achieved by fine-tuning the model on targeted datasets with minimal architectural modifications. This adaptability clearly distinguishes our method from existing approaches, which typically require extensive re-engineering during test-time optimization to accommodate such specialized scenarios.

Runtime and Memory. As shown in Tab. B, we evaluate inference runtime and peak GPU memory usage of the feature backbone when processing varying numbers of input frames. Measurements are conducted using a single NVIDIA H100 GPU with flash attention v3 [49]. Images have a resolution of 336×518.

We focus on the cost associated with the feature backbone since users may select different branch combinations depending on their specific requirements and available resources. The camera head is lightweight, typically accounting for approximately 5% of the runtime and about 2% of the GPU memory used by the feature backbone. A DPT head uses an average of 0.03 seconds and 0.2 GB of GPU memory per frame.

When GPU memory is sufficient, multiple frames can be processed efficiently in a single forward pass. At the same time, in our model, inter-frame relationships are handled only within the feature backbone, and the DPT heads make independent predictions per frame. Therefore, users constrained by GPU resources may perform predictions frame by frame. We leave this trade-off to the user's discretion.

We recognize that a naive implementation of global self-attention can be highly memory-intensive with a large number of tokens. Savings or accelerations can be achieved by employing techniques used in large language model (LLM) deployments. For instance, Fast3R [73] employs Tensor Parallelism to accelerate inference with multiple GPUs, which can be directly applied to our model.

Patchifying. As discussed in the main paper, we have explored the method of "patchifying" images into tokens by utilizing either a 14×14 convolutional layer or a pretrained DINOv2 model. Empirical results indicate that the DINOv2 model provides better performance; moreover, it ensures much more stable training, particularly in the initial stages. The DINOv2 model is also less sensitive to variations in hyperparameters such as learning rate or momentum. Consequently, we have chosen DINOv2 as the default method for patchifying in our model.

Differentiable BA. We also explored the idea of using differentiable bundle adjustment as in VGGSfM [66]. In small-scale preliminary experiments, differentiable BA demonstrated promising performance. However, a bottleneck is its computational cost during training. Enabling differentiable BA in PyTorch using Theseus [42] typically makes each training step roughly 4 times slower, which is expensive for large-scale training. While customizing a framework to expedite training could be a potential solution, it falls outside the scope of this work. Thus, we opted not to include differentiable BA in this work, but we recognize it as a promising direction for large-scale unsupervised training, as it can serve as an effective supervision signal in scenarios lacking explicit 3D annotations.

Single-view Reconstruction. Unlike systems like DUSt3R and MASt3R that have to duplicate an image to create a pair, our model architecture inherently supports the input of a single image. In this case, global attention simply transitions to frame-wise attention. Although our model was not explicitly trained for single-view reconstruction, it demonstrates surprisingly good results. Some examples can be found in Fig. A. We strongly encourage trying our demo for better visualization.

Normalizing Prediction. As discussed in Appendix B, our approach normalizes the ground truth using the average Euclidean distance of the 3D points. While some methods, such as DUSt3R, also apply such normalization to network predictions, our findings suggest that it is neither necessary for convergence nor advantageous for final model performance. Furthermore, it tends to introduce additional instability during the training phase.

References

[1] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54 (10):105–112, 2011. 4

[2] Mica Arie-Nachimson, Shahar Z Kovalsky, Ira Kemelmacher-Shlizerman, Amit Singer, and Ronen Basri. Global motion estimation from point matches. In *2012 Second international conference on 3D imaging, modeling, processing, visualization & transmission*, pages 81–88. IEEE, 2012. 4

[3] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6836–6846, 2021. 4

[4] 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 *ECCV*, 2024. 4

[5] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020. 3

[6] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 2

[7] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proc. ICCV*, 2021. 4

[8] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022. 4

[9] David J Crandall, Andrew Owens, Noah Snavely, and Daniel P Huttenlocher. Sfm with mrfs: Discrete-continuous optimization for large-scale structure from motion. *IEEE transactions on pattern analysis and machine intelligence*, 35(12):2841–2853, 2012. 4

[10] Hainan Cui, Xiang Gao, Shuhan Shen, and Zhanyi Hu. Hsfm: Hybrid structure-from-motion. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1212–1221, 2017.

[11] Zhaopeng Cui and Ping Tan. Global structure-from-motion by similarity averaging. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 864–872, 2015.

[12] Zhaopeng Cui, Nianjuan Jiang, Chengzhou Tang, and Ping Tan. Linear global translation estimation with feature tracks. *arXiv preprint arXiv:1503.01832*, 2015. 4

[13] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 2

[14] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 2

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*, 2019. 3

[16] 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 16×16 words: Transformers for image recognition at scale. In *Proc. ICLR*, 2021. 3

[17] 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*, 2409.19152, 2024. 2

[18] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Roma: Robust dense feature matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19790–19800, 2024. 2

[19] Jan-Michael Frahm, Pierre Fite-Georgel, David Gallup, Tim Johnson, Rahul Raguram, Changchang Wu, Yi-Hung Jen, Enrique Dunn, Brian Clipp, Svetlana Lazebnik, et al. Building rome on a cloudless day. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11*, pages 368–381. Springer, 2010. 4

[20] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, Thomas Kipf, Abhijit Kundu, Dmitry Lagun, Issam Laradji, Hsueh-Ti (Derek) Liu, Henning Meyer, Yishu Miao, Derek Nowrouzezahrai, Cengiz Oztireli, Etienne Pot, Noha Radwan, Daniel Rebain, Sara Sabour, Mehdi S. M. Sajjadi, Matan Sela, Vincent Sitzmann, Austin Stone, Deqing Sun, Suhani Vora, Ziyu Wang, Tianhao Wu, Kwang Moo Yi, Fangcheng Zhong, and Andrea Tagliasacchi. Kubric: a scalable dataset generator. In *Proc. CVPR*, 2022. 2

[21] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, 2004. 4

[22] Xingyi He, Jiaming Sun, Yifan Wang, Sida Peng, Qixing Huang, Hujun Bao, and Xiaowei Zhou. Detector-free structure from motion. In *arxiv*, 2023. 4

[23] Alex Henry, Prudhvi Raj Dachapally, Shubham Pawar, and Yuxuan Chen. Query-key normalization for transformers. *arXiv preprint arXiv:2010.04245*, 2020. 1, 4

[24] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[25] Nianjuan Jiang, Zhaopeng Cui, and Ping Tan. A global linear method for camera pose registration. In *Proceedings of the IEEE international conference on computer vision*, pages 481–488, 2013. 4

[26] 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, 2021. 2, 4

[27] Nikita Karaev, Ignacio Rocco, Ben Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. CoTracker: It is better to track together. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024. 1

[28] Alex Kendall and Yarin Gal. What uncertainties do we need in Bayesian deep learning for computer vision? *Proc. NeurIPS*, 2017. 1

[29] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. *arXiv preprint arXiv:2406.09756*, 2024. 2, 4

[30] 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. 2

[31] Amy Lin, Jason Y Zhang, Deva Ramanan, and Shubham Tulsiani. Relpose++: Recovering 6d poses from sparse-view observations. *arXiv preprint arXiv:2305.04926*, 2023. 4

[32] Philipp Lindenberger, Paul-Edouard Sarlin, Viktor Larsson, and Marc Pollefeys. Pixel-perfect structure-from-motion with featuremetric refinement. *arXiv.cs*, abs/2108.08291, 2021. 2, 4

[33] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22160–22169, 2024. 2

[34] Manuel Lopez-Antequera, Pau Gargallo, Markus Hofinger, Samuel Rota BulÃ², Yubin Kuang, and Peter Kontschieder. Mapillary planet-scale depth dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2

[35] Pierre Moulon, Pascal Monasse, and Renaud Marlet. Global fusion of relative motions for robust, accurate and scalable structure from motion. In *Proceedings of the IEEE international conference on computer vision*, pages 3248–3255, 2013. 4

[36] David Novotný, Diane Larlus, and Andrea Vedaldi. Learning 3D object categories by looking around them. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2017. 1

[37] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. 1

[38] Onur Ozyesil and Amit Singer. Robust camera location estimation by convex programming. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2674–2683, 2015. 4

[39] Linfei Pan, Daniel Barath, Marc Pollefeys, and Johannes Lutz Schönberger. Global Structure-from-Motion Revisited. In *European Conference on Computer Vision (ECCV)*, 2024. 4

[40] Xiaqing Pan, Nicholas Charron, Yongqian Yang, Scott Peters, Thomas Whelan, Chen Kong, Omkar Parkhi, Richard

Newcombe, and Yuheng (Carl) Ren. Aria digital twin: A new benchmark dataset for egocentric 3d machine perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20133–20143, 2023. 2

[41] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 4

[42] Luis Pineda, Taosha Fan, Maurizio Monge, Shobha Venkataraman, Paloma Sodhi, Ricky TQ Chen, Joseph Ortiz, Daniel DeTone, Austin Wang, Stuart Anderson, et al. Theseus: A library for differentiable nonlinear optimization. *Advances in Neural Information Processing Systems*, 35:3801–3818, 2022. 5

[43] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021. 1

[44] 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 *Proc. ICCV*, 2021. 2

[45] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *International Conference on Computer Vision (ICCV) 2021*, 2021. 2

[46] Rother. Linear multiview reconstruction of points, lines, planes and cameras using a reference plane. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 1210–1217. IEEE, 2003. 4

[47] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 4

[48] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proc. CVPR*, 2016. 2

[49] Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. *Advances in Neural Information Processing Systems*, 37: 68658–68685, 2024. 4

[50] Samarth Sinha, Jason Y Zhang, Andrea Tagliasacchi, Igor Gilitschenski, and David B Lindell. Sparsepose: Sparse-view camera pose regression and refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21349–21359, 2023. 4

[51] Cameron Smith, David Charatan, Ayush Tewari, and Vincent Sitzmann. Flowmap: High-quality camera poses, intrinsics, and depth via gradient descent. *arXiv preprint arXiv:2404.15259*, 2024. 4

[52] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006. 4

[53] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 2

[54] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8922–8931, 2021. 2

[55] Chris Sweeney, Torsten Sattler, Tobias Hollerer, Matthew Turk, and Marc Pollefeys. Optimizing the viewing graph for structure-from-motion. In *Proceedings of the IEEE international conference on computer vision*, pages 801–809, 2015. 4

[56] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: Training home assistants to rearrange their habitat. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 2

[57] Chengzhou Tang and Ping Tan. Ba-net: Dense bundle adjustment network. *arXiv preprint arXiv:1806.04807*, 2018. 4

[58] Zachary Teed and Jia Deng. Deepv2d: Video to depth with differentiable structure from motion. *arXiv preprint arXiv:1812.04605*, 2018.

[59] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *Advances in neural information processing systems*, 34:16558–16569, 2021. 4

[60] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021. 4

[61] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 32–42, 2021. 1, 4

[62] Benjamin Ummenhofer, Huizhong Zhou, Jonas Uhrig, Nikolaus Mayer, Eddy Ilg, Alexey Dosovitskiy, and Thomas Brox. Demon: Depth and motion network for learning monocular stereo. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5038–5047, 2017. 4

[63] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[64] Jianyuan Wang, Yiran Zhong, Yuchao Dai, Stan Birchfield, Kaihao Zhang, Nikolai Smolyanskiy, and Hongdong Li. Deep two-view structure-from-motion revisited. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 8953–8962, 2021. 4

[65] Jianyuan Wang, Christian Rupprecht, and David Novotny. Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9773–9783, 2023. 4

[66] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. VGGSfM: visual geometry grounded deep structure from motion. In *Proc. CVPR*, 2024. 2, 3, 4, 5

[67] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. DUSt3R: Geometric 3D vision made easy. In *Proc. CVPR*, 2024. 1, 2, 4

[68] Xingkui Wei, Yinda Zhang, Zhuwen Li, Yanwei Fu, and Xiangyang Xue. Deepsfm: Structure from motion via deep bundle adjustment. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 230–247. Springer, 2020. 4

[69] Changchang Wu. Towards linear-time incremental structure from motion. In *2013 International Conference on 3D Vision-3DV 2013*, pages 127–134. IEEE, 2013. 4

[70] Hongchi Xia, Yang Fu, Sifei Liu, and Xiaolong Wang. Rgbd objects in the wild: Scaling real-world 3d object learning from rgb-d videos, 2024. 2

[71] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34: 12077–12090, 2021. 4

[72] Fei Xie, Chunyu Wang, Guangting Wang, Yue Cao, Wankou Yang, and Wenjun Zeng. Correlation-aware deep tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8751–8760, 2022. 4

[73] Jianing Yang, Alexander Sax, Kevin J Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. *arXiv preprint arXiv:2501.13928*, 2025. 4

[74] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv:2406.09414*, 2024. 1

[75] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1790–1799, 2020. 2

[76] Shuangfei Zhai, Tatiana Likhomanenko, Etai Littwin, Dan Busbridge, Jason Ramapuram, Yizhe Zhang, Jiatao Gu, and Joshua M Susskind. Stabilizing transformer training by preventing attention entropy collapse. In *International Conference on Machine Learning*, pages 40770–40803. PMLR, 2023. 4

[77] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. MonST3R: a simple approach for estimating geometry in the presence of motion. *arXiv*, 2410.03825, 2024. 1

[78] Jason Y Zhang, Deva Ramanan, and Shubham Tulsiani. Relpose: Predicting probabilistic relative rotation for single objects in the wild. In *ECCV*, pages 592–611. Springer, 2022. 4

[79] Jason Y Zhang, Amy Lin, Moneish Kumar, Tzu-Hsuan Yang, Deva Ramanan, and Shubham Tulsiani. Cameras as rays: Pose estimation via ray diffusion. In *International Conference on Learning Representations (ICLR)*, 2024. 4

[80] Yang Zheng, Adam W. Harley, Bokui Shen, Gordon Wetzstein, and Leonidas J. Guibas. Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In *ICCV*, 2023. 2

[81] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1851–1858, 2017. 4