MoVieS: Motion-Aware 4D Dynamic View Synthesis in One Second

Supplementary Material

A. Implementation Details

A.1. Dataset Details

Detailed information about the training datasets used in this work is provided in Table S.1. The majority of training datasets used in our experiments are synthetic, providing rich annotations such as pixel-aligned depth maps and accurate camera intrinsics and extrinsics. Ground-truth 3D point trajectories are also available in PointOdyssey [27] and DynamicReplica [7]. On the other hand, Stereo4D [6] is a large-scale dataset constructed from YouTube stereo videos and annotated using pretrained foundation models [5, 21]. Despite being partially noisy, its diversity and scale offer strong generalization benefits.

The pretrained backbone, VGGT [22], requires all 3D scenes to be normalized to a unit scale on average, which in turn necessitates depth maps and camera extrinsics to be defined in a consistent metric space. It is satisfied by all synthetic datasets and Stereo4D, whose camera poses and depths are metric-aligned by construction. However, RealEstate10K [28] only provides relative camera parameters estimated via COLMAP [19], resulting in an unknown global scale. To address this issue, we re-estimate both depth maps and camera extrinsics using recent foundation models, including Video Depth Anything [3], Depth Pro [1] and MegaSaM [13], to recover aligned geometry across frames.

A.2. Curriculum Training

We observed that training MoVieS is particularly unstable: the training loss often fluctuates abruptly, and gradients are prone to becoming `None`. It may arise from sparse annotations and the heterogeneous nature of training datasets, which mix datasets from various sources with differing domains (e.g., indoor *vs*. outdoor, real *vs*. synthetic), camera FoV, recording frame rates, etc.

Thanks to the versatile design of MoVieS, we employ a curriculum strategy that gradually increases the training complexity. It begins by pretraining the model on low-resolution (224×224) static datasets with only depth and photometric losses, then introduces dynamic datasets along with motion supervision. We found that static datasets play a crucial role in stabilizing training for dynamic scenes, without which the loss would be highly unstable. Since modeling dynamic scenes requires reconstructing a set of 3DGS for each query time, which results in high GPU memory usage, we start by training on 5 input views for dynamic scenes and then expand to 13 views for fine-tuning. Finally, the training resolution is increased to 518. Similar to VGGT [22], in the last training stage, we randomly sample the frame number from $2 \sim 13$ and the aspect ratio from $0.5 \sim 2$, with the largest side fixed to 518.

A.3. Training Details

Image encoder, feature backbone, and depth head of MoVieS are initialized from VGGT [22]. Motion head is initialized from its pointmap head. Remaining components, such as the splatter head and camera/time embeddings, are trained from scratch. We use AdamW optimizer [15] with a weight decay of `0.05`, and adopt a cosine learning rate scheduler [14] with linear warm-up for all curriculum training stages. For static pretraining and dynamic scenes with 5 and 13 input views at a resolution of 224×224, we use learning rates of `4e-4`, `4e-4` and `4e-5`, respectively, with a batch size of `256`. Training rates for parameters initialized from VGGT are multiplied by `0.1`. With `32` H20 GPUs, training takes about `2` days for static and then dynamic scenes with 5 views, and `2` days for 13 views. MoVieS is then finetuned on 518×518 videos with 13 frames using a learning rate of `1e-5`, which takes around `1` day. To improve memory and computation efficiency, several techniques such as `gsplat` [26] rendering backend, DeepSpeed [18], gradient checkpointing [4], gradient accumulation, and `bf16` mixed precision are also adopted.

For the multi-task training objective, we did not manually tune the relative weights between different loss terms. Instead, the weights were set to bring the numerical values of the losses into roughly similar ranges: $\lambda_{\text{d}} = \lambda_{\text{r}} = \lambda_{\text{pt}} = 1$ and $\lambda_{\text{m}} = \lambda_{\text{dist}} = 10$.

B. Limitations and Future Work

Although MoVieS achieves competitive performance with inference speeds that are orders of magnitude faster than optimization-based methods, there remains a noticeable gap in reconstruction quality. This gap arises partly because many optimization-based approaches benefit from multiple pretrained models that preprocess input videos to provide richer and more accurate cues. Incorporating such richer prior knowledge directly into MoVieS represents a promising direction for future work to further improve reconstruction fidelity and robustness.

Currently, MoVieS depends on off-the-shelf tools for camera parameter estimation, which adds an external dependency and may limit end-to-end optimization. Seamlessly integrating camera pose estimation within the MoVieS pipeline could simplify the overall workflow, reduce error accumulation, and enhance adaptability to di-

Table S.1. **Training Datasets**. Eight datasets from diverse sources are utilized to train MoVieS at scale. "#Repeat" denotes dataset duplication count during integration to balance their contributions.

Dataset	Dynamic?	Depth?	Tracking?	Real?	#Scenes	#Frames	#Repeat
RealEstate10K [28]	✗	✗	✗	✔	70K	6.36M	1×
TartanAir [24]	✗	✔	✗	✗	0.4K	0.49M	100×
MatrixCity [11]	✗	✔	✗	✗	4.5K	0.31M	10×
PointOdyssey [27]	✔	✔	✔	✗	0.1K	0.18M	1000×
DynamicReplica [7]	✔	✔	✔	✗	0.5K	0.26M	100×
Spring [16]	✔	✔	✗	✗	0.03K	0.003M	2000×
VKITTI2 [2]	✔	✔	✗	✗	0.1K	0.03M	500×
Stereo4D [6]	✔	✔	✔	✔	98K	19.6M	1×

verse scenarios.

Moreover, scaling MoVieS to handle long videos and achieve high-resolution rendering remains a challenge. The computational cost and memory demand grow significantly with scene complexity and resolution, which constrains practical deployment. Developing more compact and efficient dynamic scene representations, possibly through novel model architectures or sparse encoding strategies, is essential to push 4D reconstruction towards real-world applications.

Addressing these challenges will not only bridge the performance gap but also unlock the full potential of fast and high-quality dynamic scene reconstruction.

C. License Information

We employ several open-source implementations in our experimental comparisons, including: (1) DepthSplat [25][1] (MIT License), (2) Splatter-a-Video [20][2] (Apache License), (3) Shape-of-Motion [23][3] (MIT License), (4) MoSca [10][4] (MIT License), (5) BootsTAPIR [5][5] (Apache License), (6) CoTracker3 [8][6] (Creative Commons Attribution-NonCommercial 4.0 International Public License), and (7) SpatialTracker [12][7] (Attribution-NonCommercial 4.0 International).

Datasets from diverse sources are utilized to train MoVieS, including: (1) RealEstate10K [28][8] (Creative Commons Attribution 4.0 International License), (2) TartanAir [24][9] (Creative Commons Attribution 4.0 International License), (3) MatrixCity [11][10] (Apache License), (4) PointOdyssey [27][11] (MIT License), (5) DynamicReplica [7][12] (Attribution-NonCommercial 4.0 International License), (6) Spring [16][13] (CC BY 4.0), (7) VKITTI2 [2][14] (Creative Commons Attribution-NonCommercial-ShareAlike 3.0), and (8) Stereo4D [6][15] (CC0 1.0 Universal).

D. Broader Impact

MoVieS provides substantial advantages for fields including robotics simulation, AR/VR, autonomous driving, and digital twins by enabling fast and generalizable dynamic scene understanding. Its ability to efficiently reconstruct dynamic environments can accelerate innovation and improve system performance in these areas. However, such powerful technology also raises risks related to the unauthorized generation of content and potential privacy violations. To mitigate these risks, it is essential to establish clear ethical guidelines and implement appropriate regulatory measures to ensure responsible and safe usage.

E. More Visualization Results

We provide more visualization results on novel view synthesis, 3D point tracking, scene flow estimation, and dynamic object segmentation in Figure S.1, S.2, S.3 and S.4 respectively. Qualitative visualizations of novel view synthesis and 3D point tracking are conducted on the DAVIS dataset [17] and TAPVid-3D [9] respectively, which are not included in the training datasets. Other visualization results are conducted on the test set of the curated datasets to evaluate the generalizability of MoVieS. XYZ values in the 3D space of motion maps are normalized to $[0, 1]$ and treated as RGB channels for visualization.

[1] https://github.com/cvg/depthsplat/tree/main
[2] https://github.com/SunYangtian/Splatter_A_Video
[3] https://github.com/vye16/shape-of-motion/
[4] https://github.com/JiahuiLei/MoSca
[5] https://github.com/google-deepmind/tapnet
[6] https://github.com/facebookresearch/co-tracker
[7] https://github.com/henry123-boy/SpaTracker
[8] https://google.github.io/realestate10k/
[9] https://theairlab.org/tartanair-dataset/
[10] https://city-super.github.io/matrixcity/
[11] https://pointodyssey.com/
[12] https://github.com/facebookresearch/dynamic_stereo
[13] https://spring-benchmark.org/
[14] https://europe.naverlabs.com/research/computer-vision/proxy-virtual-worlds-vkitti-2/
[15] https://stereo4d.github.io/

Figure S.1. **Qualitative results of novel view synthesis on the DAVIS dataset** [17]. Camera poses are estimated by MegaSaM [13]. "Motion*" denotes the 3D pixel displacements between the "Target Time" frame with respect to the "Target Viewpoint" frame. The "Novel View Synthesis" results show the reconstructed scene at the target time from the target viewpoint.

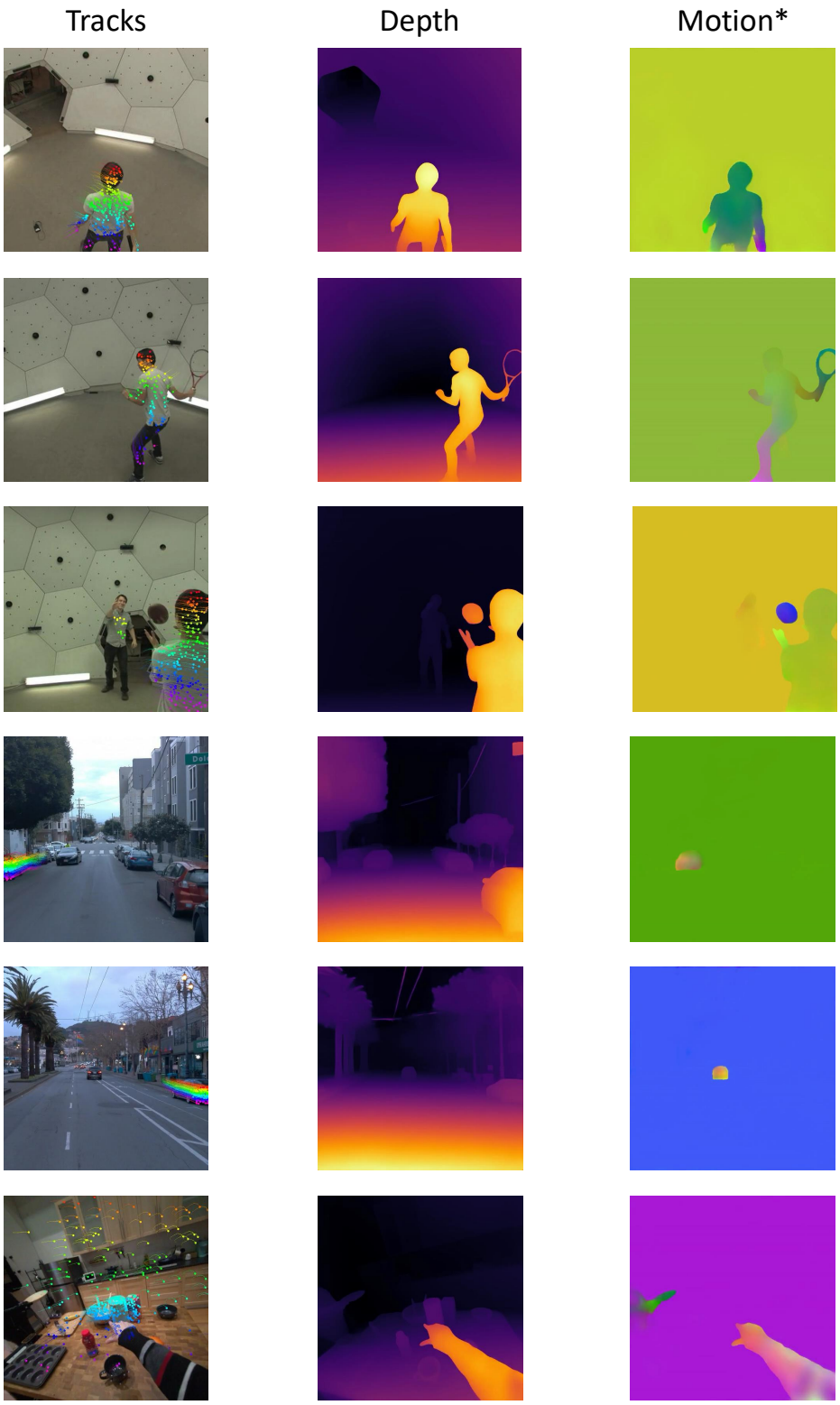

Figure S.2. **Qualitative results for 3D point tracking**. "Motion*" means the 3D points' movements of the input frame with respect to the first one. Points in the 3D space are projected to the image space for 2D point tracking visualization.

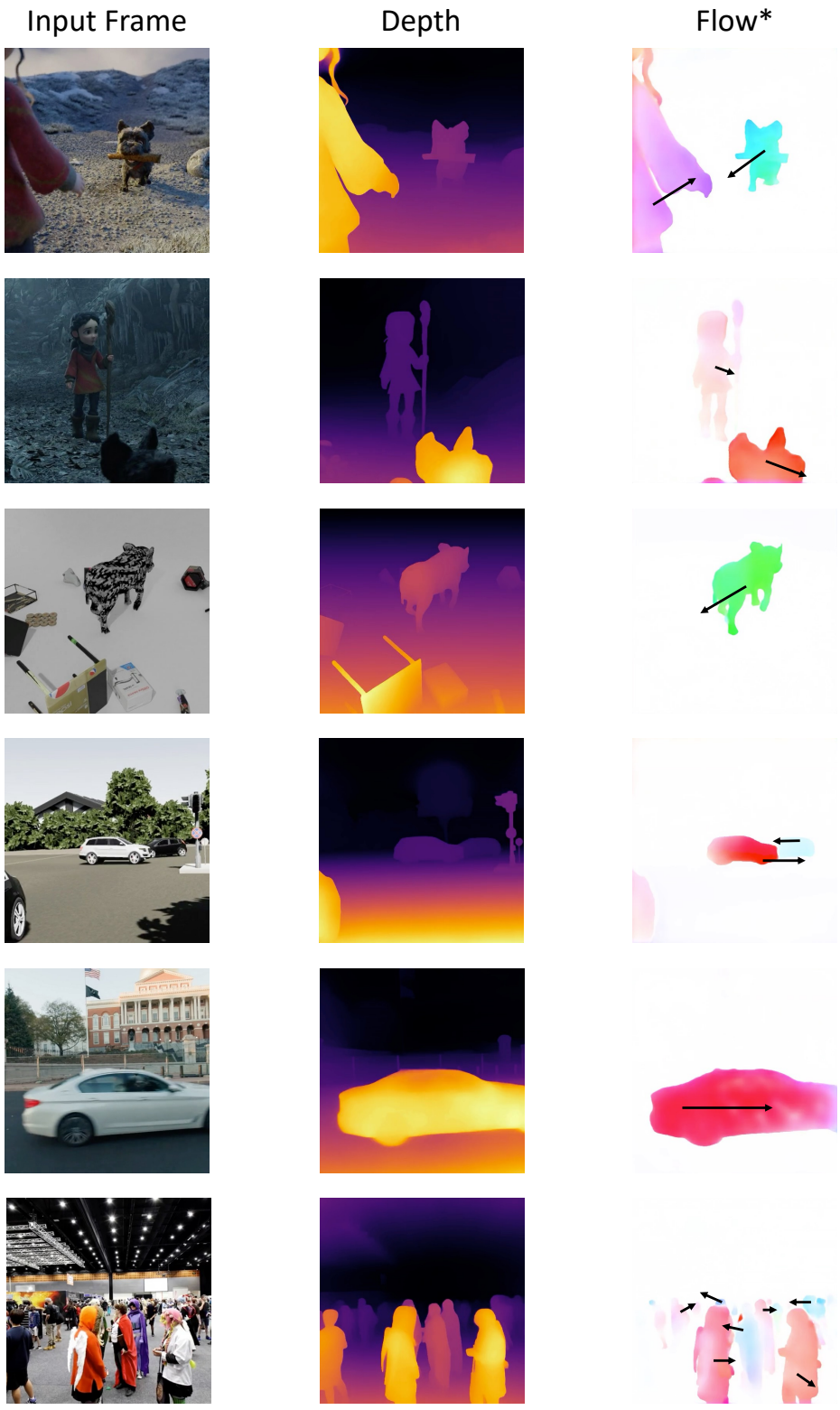

Figure S.3. **Qualitative results for scene flow estimation**. "Flow*" means the optical flow of the input pixels with respect to the first frame, and is obtained by projecting the estimated motion maps in the 3D space to the camera space. Each color of optical flow means a specific 2D direction and arrows on pictures roughly mark the directions.

Figure S.4. **Qualitative results for moving object segmentation**. Masks for moving parts, which are filtered from the values of estimated motion maps, are highlighted across frames in light red. Regions with high motion values are regarded as moving parts.

References

[1] Aleksei Bochkovskii, AmaÃĢl Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. In *International Conference on Learning Representations (ICLR)*, 2024. 1

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

[3] Sili Chen, Hengkai Guo, Shengnan Zhu, Feihu Zhang, Zilong Huang, Jiashi Feng, and Bingyi Kang. Video depth anything: Consistent depth estimation for super-long videos. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 1

[4] Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. Training deep nets with sublinear memory cost. *arXiv preprint arXiv:1604.06174*, 2016. 1

[5] Carl Doersch, Pauline Luc, Yi Yang, Dilara Gokay, Skanda Koppula, Ankush Gupta, Joseph Heyward, Ignacio Rocco, Ross Goroshin, João Carreira, et al. Bootstap: Bootstrapped training for tracking-any-point. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2024. 1, 2

[6] Linyi Jin, Richard Tucker, Zhengqi Li, David Fouhey, Noah Snavely, and Aleksander Holynski. Stereo4d: Learning how things move in 3d from internet stereo videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 1, 2

[7] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Dynamicstereo: Consistent dynamic depth from stereo videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2

[8] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. In *European Conference on Computer Vision (ECCV)*, 2024. 2

[9] Skanda Koppula, Ignacio Rocco, Yi Yang, Joe Heyward, João Carreira, Andrew Zisserman, Gabriel Brostow, and Carl Doersch. Tapvid-3d: A benchmark for tracking any point in 3d. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 2

[10] Jiahui Lei, Yijia Weng, Adam Harley, Leonidas Guibas, and Kostas Daniilidis. Mosca: Dynamic gaussian fusion from casual videos via 4d motion scaffolds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2

[11] Yixuan Li, Lihan Jiang, Linning Xu, Yuanbo Xiangli, Zhenzhi Wang, Dahua Lin, and Bo Dai. Matrixcity: A large-scale city dataset for city-scale neural rendering and beyond. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 2

[12] Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[13] Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast, and robust structure and motion from casual dynamic videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 1, 3

[14] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations (ICLR)*, 2016. 1

[15] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2018. 1

[16] Lukas Mehl, Jenny Schmalfuss, Azin Jahedi, Yaroslava Nalivayko, and Andrés Bruhn. Spring: A high-resolution high-detail dataset and benchmark for scene flow, optical flow and stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4981–4991, 2023. 2

[17] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 3

[18] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 2020. 1

[19] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[20] Yang-Tian Sun, Yihua Huang, Lin Ma, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Splatter a video: Video gaussian representation for versatile processing. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 2

[21] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European Conference on Computer Vision (ECCV)*, 2020. 1

[22] 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 (CVPR)*, 2025. 1

[23] Qianqian Wang, Vickie Ye, Hang Gao, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4d reconstruction from a single video. In *International Conference on Computer Vision (ICCV)*, 2025. 2

[24] 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 *International Conference on Intelligent Robots and Systems (IROS)*, 2020. 2

[25] Haofei Xu, Songyou Peng, Fangjinhua Wang, Hermann Blum, Daniel Barath, Andreas Geiger, and Marc Pollefeys. Depthsplat: Connecting gaussian splatting and depth. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2

[26] Vickie Ye, Ruilong Li, Justin Kerr, Matias Turkulainen, Brent Yi, Zhuoyang Pan, Otto Seiskari, Jianbo Ye, Jeffrey Hu, Matthew Tancik, et al. gsplat: An open-source library for

gaussian splatting. *Journal of Machine Learning Research (JMLR)*, 2025. 1

[27] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 1, 2

[28] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: learning view synthesis using multiplane images. *Transactions on Graphics (TOG)*, 2018. 1, 2