Spring: A High-Resolution High-Detail Dataset and Benchmark for Scene Flow, Optical Flow and Stereo – Supplementary Material –

In the following, we provide additional information regarding the Spring dataset and benchmark. In this context, we present further example sequences and visualizations of the methods we evaluated, we continue the discussion of results from the main paper, list additional evaluation results excluding the sky region, and show screenshots of the benchmark website.

1. Further examples of the Spring dataset

In Fig. 1, we show further example sequences of the Spring dataset that illustrate its *wide variety of content*. For each sequence, we show the left and right image of the stereo camera, the corresponding left and right disparity, the change of the left and right disparity both in forward and backward temporal direction, as well as the left and right optical flow also both in forward and backward temporal direction, accordingly.

2. Visual benchmark results

Moreover, for the stereo, optical flow and scene flow tasks, we show disparity/flow and error visualizations of the methods we evaluated in Figs. 2 to 4.

3. Further discussion of results

Optical Flow. For optical flow, we can see that the handling of high-resolution inputs plays an important role for the performance on our benchmark. Many of the evaluated networks estimate the optical flow on a lower resolution, followed by a learned upsampling; FlowFormer [2], RAFT [11] and GMA [6] work on 1/8th of the original resolution, GMFlow [14] on 1/4th. The best-performing MS-RAFT+ [4, 5] works on an even higher resolution, 1/2 of the original resolution, also followed by a learned upsampling. In general, methods with learned upsampling lead the benchmark, a strategy closely related to FlowNet2 [3] ranking third. Their architecture consists of modules predicting optical flow on 1/4th of the original resolution, but then uses a fusion module that given nearest-neighbor upsampled inputs predicts results on the original resolution. In contrast, the coarse-to-fine pyramid strategy of SPyNet [9] as well as the purely bilinear $4\times$ upsampling of PWCNet [10] did not yield as good results as the aforementioned strategies.

When comparing EPE results from our benchmark to EPE results from the Sintel benchmark, it is noticeable that numbers on our benchmark are on a lower level. We attribute this mainly to the fact that the Sintel dataset has a focus on action sequences with very strong motion, while Spring addresses high-resolution and high-detail content. Although there are also several high-speed scenes in Spring, they cover a smaller part of the dataset. Further, with the super-resolution ground truth, Spring uses a more permissive evaluation methodology than Sintel.

Stereo. In case of stereo, one can observe that the best performing methods on the Spring benchmark operate on moderately subsampled versions of the input images – *i.e.* typically 1/3 or 1/4 of the original resolution – while they rely on hierarchical concepts at the same time. More precisely, ACVNet [13] uses three-level adaptive patch matching with attention-based feature concatenation, RAFT-Stereo [7] exploits a four-level correlation pyramid, multi-level recurrent update operators and a three level coarse-to-fine estimation scheme, and LEAStereo [1] learns a compact network with 2-level feature extractor and 3-level matching module based on a neural architecture search. GA-Net [15] which ranks last in the Spring benchmark is the only network that directly operates on the original resolution. While it considers guided aggregation layers, however, it does not exploit hierarchical concepts. From all considered methods, ACVNet performs best. This can not only be seen from the corresponding table in the main paper, but also from a visual comparison of the results in Fig. 2. While it shows slight upsampling artefacts, ACVNet seems most robust regrading the background estimation, potentially due to the attention-based feature concatenation.

Scene Flow. In case of scene flow, two of the considered approaches (RAFT-3D [12], M-FUSE [8]) rely on a RGB-D setting. This in turn requires the pre-computation of stereo results before estimating the scene flow. In contrast, CamliFlow seeks to better preserve the 3D structure of the scene by integrating LiDAR input that is converted to point clouds. However, in case of stereo input, CamliFlow also has to rely on external stereo results before constructing these point clouds. Interestingly, all three approaches show problems with different components of the scene flow. M-FUSE that extends RAFT-3D by considering temporal information and using LEAStereo as a stereo baseline shows only moderately accurate results for the optical flow – although the stereo input seems be of good quality. In contrast, RAFT-3D that relies on GA-Net shows larger errors in the stereo estimation that eventually propagate to the overall scene flow. Finally, CamliFlow has severe problems obtaining useful disparity estimates for the second frame pair, most probably due to the same difficulties as RAFT-3D with the underlying GA-Net approach. Another observation that might explain the comparably poor performance of CamliFlow is its dedicated background estimation module. Since it is trained on Cityscapes and KITTI it is likely not to generalize to other type of data. In terms of overall accuracy, M-FUSE slightly outperforms MS-RAFT. In contrast, CamliFlow gives significantly worse results. As in the the stereo case, these findings are not only reflected in the corresponding results in the main paper. They can also be seen from the visual comparison in Fig. 4.

4. Additional results for non-sky regions

As described in the main paper, we provide additional evaluation results that consider *non-sky pixels only*. The corresponding rankings for stereo, optical flow and scene flow can be found in Tabs. 1 to 3. As expected, the difficulty decreases such that the overall errors are lower, resulting in a few ranking order changes. The largest decrease is noticeable for the stereo benchmark, showing that the sky regions are particularly difficult for the current set of methods. At the same time, it is expected that future stereo methods trained on datasets with sky regions will perform better.

5. Screenshots of the benchmark website

Finally, in Figs. 5 to 12, *screenshots* of the Spring benchmark website are shown. These screenshots include the benchmark start page (Fig. 5), the current benchmark rankings for stereo (Fig. 6), optical flow (Fig. 7), and scene flow (Fig. 8), the benchmark evaluation sites of the three leading methods, i.e. ACVNet (Fig. 9), MS-RAFT+ (Fig. 10), and M-FUSE (Fig. 11), as well as the benchmark page to sign up for an evaluation account (Fig. 12).

References

[1] Xuelian Cheng, Yiran Zhong, Mehrtash Harandi, Yuchao Dai, Xiaojun Chang, Hongdong Li, Tom Drummond, and Zongyuan Ge. Hierarchical neural architecture search for deep stereo matching. In *Proc. Conference on Neural Information Processing Systems (NeurIPS)*, pages 22158–22169, 2020. 1

[2] Zhaoyang Huang, Xiaoyu Shi, Chao Zhang, Qiang Wang, Ka Chun Cheung, Hongwei Qin, Jifeng Dai, and Hongsheng Li. FlowFormer: a transformer architecture for optical flow. In *Proc. European Conference on Computer Vision (ECCV)*, 2022. 1

[3] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. FlowNet 2.0: Evolution of optical flow estimation with deep networks. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2462–2470, 2017. 1

[4] Azin Jahedi, Maximilian Luz, Lukas Mehl, Marc Rivinius, and Andrés Bruhn. High resolution multi-scale RAFT (Robust Vision Challenge 2022). In *arXiv preprint 2210.16900*. arXiv, 2022. 1

[5] Azin Jahedi, Lukas Mehl, Marc Rivinius, and Andrés Bruhn. Multi-scale RAFT: Combining hierarchical concepts for learning-based optical flow estimation. In *Proc. IEEE International Conference on Image Processing (ICIP)*, pages 1236–1240, 2022. 1

[6] Shihao Jiang, Dylan Campbell, Yao Lu, Hongdong Li, and Richard Hartley. Learning to estimate hidden motions with global motion aggregation. In *Proc. IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 1

[7] Lahav Lipson, Zachary Teed, and Jia Deng. RAFT-Stereo: Multilevel recurrent field transforms for stereo matching. In *International Conference on 3D Vision (3DV)*, 2021. 1

[8] Lukas Mehl, Azin Jahedi, Jenny Schmalfuss, and Andrés Bruhn. M-FUSE: Multi-frame fusion for scene flow estimation. In *Proc. IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023. 1

[9] Anurag Ranjan and Michael J. Black. Optical flow estimation using a spatial pyramid network. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[10] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[11] Zachary Teed and Jia Deng. RAFT: recurrent all-pairs field transforms for optical flow. In *Proc. European Conference on Computer Vision (ECCV)*, pages 402–419, 2020. 1

[12] Zachary Teed and Jia Deng. RAFT-3D: Scene flow using rigid-motion embeddings. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8375–8384, 2021. 1

[13] Gangwei Xu, Junda Cheng, Peng Guo, and Xin Yang. Attention concatenation volume for accurate and efficient stereo matching. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1

[14] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. GMFlow: Learning optical flow via global matching. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1

[15] Feihu Zhang, Victor Prisacariu, Ruigang Yang, and Philip H. S. Torr. GA-Net: Guided aggregation net for end-to-end stereo matching. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 185–194, 2019. 1

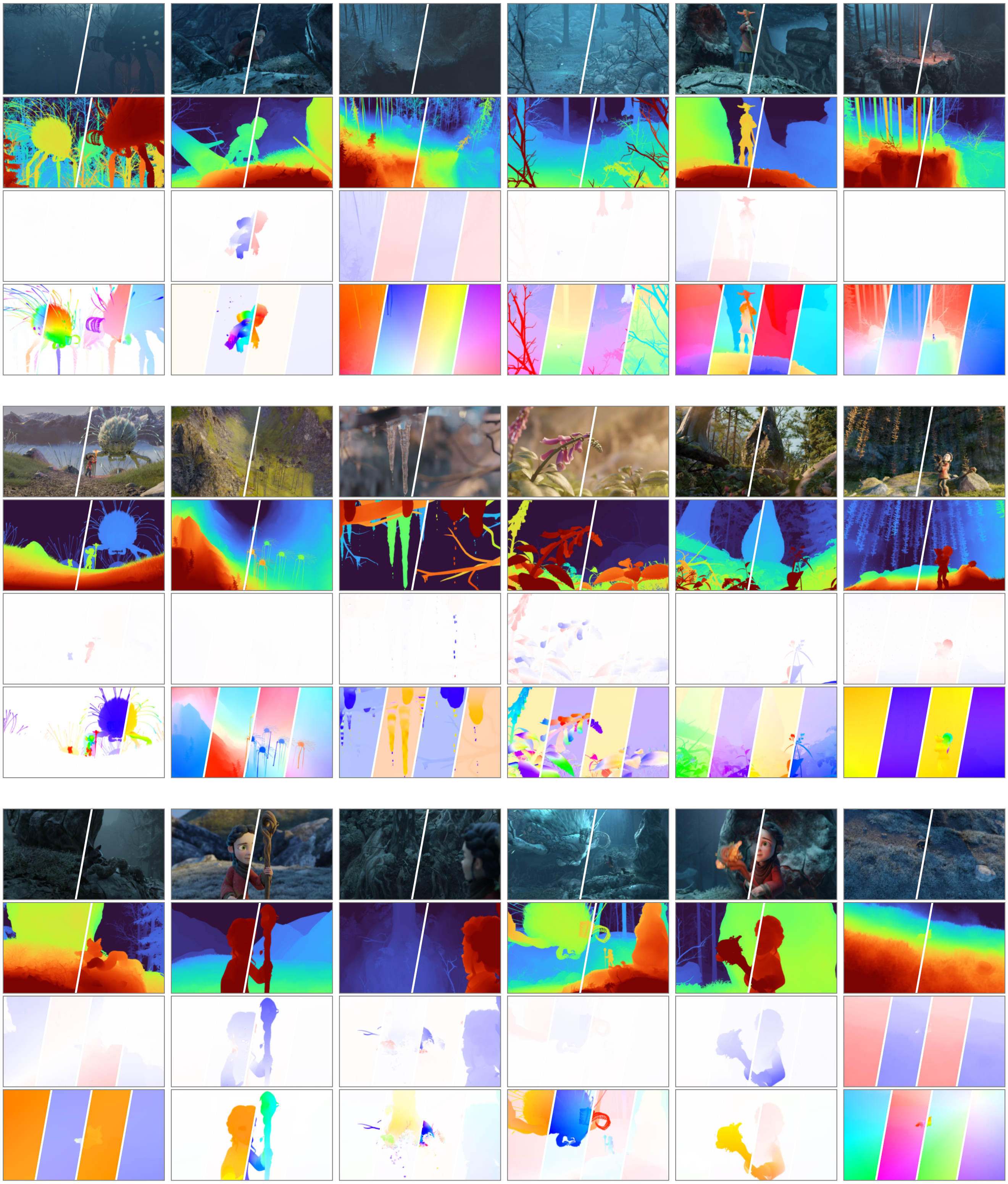

Figure 1. Example sequences from the Spring dataset. *First row*: Left and right images of the stereo camera, *second row*: Corresponding left and right disparity, *third row*: Change in disparity for forward left, backward left, forward right and backward right, *fourth row*: Optical flow visualization for forward left, backward left, forward right and backward right. Please note that we show the disparity change for visualization purposes while the dataset contains the target frame disparity.

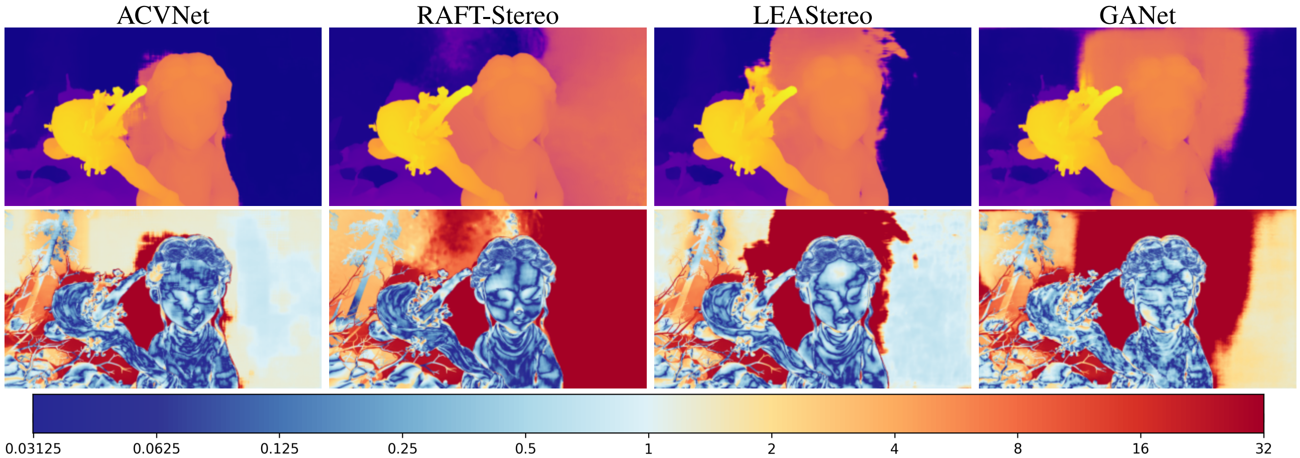

Figure 2. Visualizations from the stereo benchmark. *Top row*: predicted disparity, *bottom row*: absolute error visualization and color code.

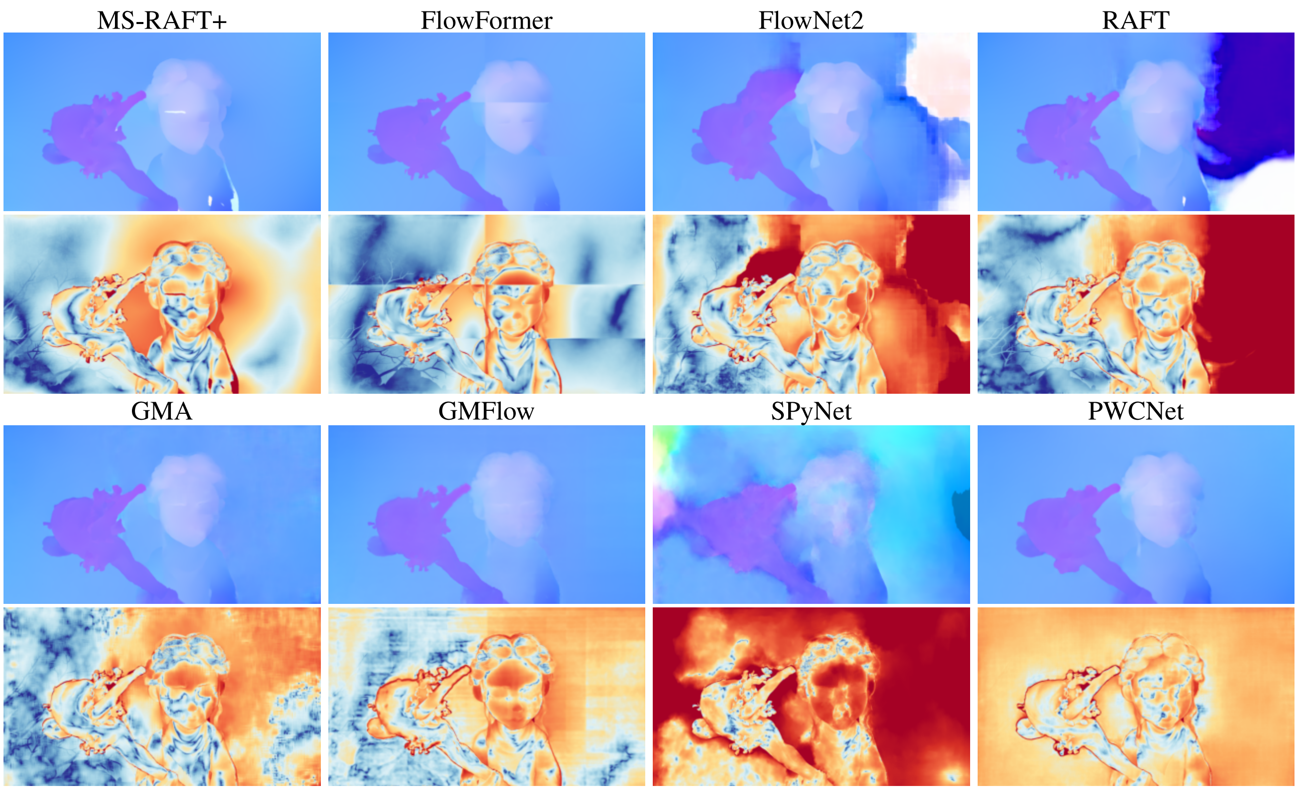

Figure 3. Visualizations from the optical flow benchmark. *Top row*: predicted optical flow, *bottom row*: EPE error visualization.

Table 1. Stereo results on our benchmark. We show additional evaluation metrics computed only on the *non-sky pixels* of the dataset.

	1px								Abs	D1
Method	total	low-det.	high-det.	matched	unmat.	s0-10	s10-40	s40+		
RAFT-Stereo	9.92	9.56	32.14	8.46	43.39	5.16	9.99	17.03	0.68	3.67
ACVNet	11.16	10.77	35.13	9.44	50.63	6.32	11.33	18.12	1.14	4.59
LEAStereo	16.73	16.36	39.65	14.91	58.50	7.63	13.84	39.39	2.44	7.42
GANet	18.42	18.03	42.16	16.51	62.09	7.32	16.41	41.48	2.59	7.77

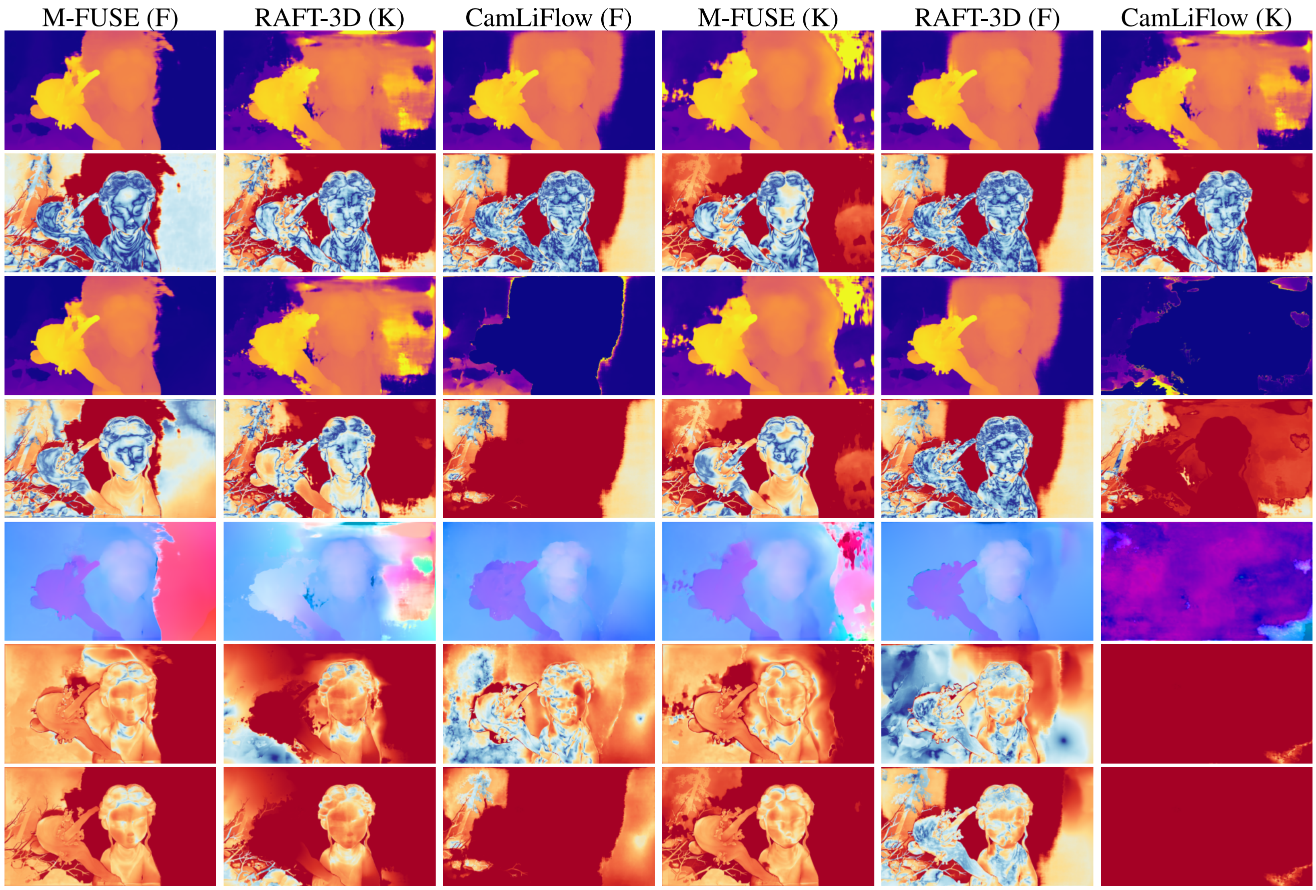

Figure 4. Visualizations from the scene flow benchmark. *From top to bottom*: predicted reference disparity, reference disparity error, predicted target disparity, target disparity error, predicted optical flow, optical flow error, combined scene flow error.

Table 2. Optical flow results on our benchmark. We show additional evaluation metrics computed only on the *non-sky pixels* of the dataset.

	1px										EPE	Fl	WAUC
Method	total	low-det.	high-det.	matched	unmat.	rigid	non-rigid	s0-10	s10-40	s40+			
MS-RAFT+	4.84	4.46	61.80	4.18	32.41	1.84	25.96	1.43	4.93	34.60	0.63	2.08	93.64
RAFT	5.25	4.86	64.30	4.47	37.74	2.15	27.08	1.75	5.15	36.39	0.71	2.09	92.28
FlowFormer	5.50	5.11	64.30	4.75	36.60	2.16	29.07	2.09	5.54	35.37	0.69	2.14	92.50
GMA	5.61	5.21	66.41	4.83	38.31	2.40	28.24	2.27	5.28	36.06	0.87	2.23	91.93
FlowNet2	6.04	5.65	64.31	5.05	47.31	2.72	29.39	1.72	5.72	45.05	0.84	2.27	91.62
GMFlow	8.95	8.50	76.64	7.65	63.10	4.94	37.24	4.01	9.66	50.34	0.94	2.75	82.98
SPyNet	25.83	25.49	77.88	24.58	77.74	21.46	56.61	19.68	23.42	87.28	3.23	8.72	70.71
PWCNet	81.57	81.57	81.76	81.37	90.07	82.07	78.09	80.57	82.09	88.82	2.25	4.17	46.40

Table 3. Scene flow results on our benchmark. We show additional evaluation metrics computed only on the *non-sky pixels* of the dataset.

	1px										SF	$1px^{D1}$	$1px^{D2}$	$1px^{Fl}$
Method	total	low-det.	high-det.	matched	unmat.	rigid	non-rigid	s0-10	s10-40	s40+				
M-FUSE (F)	31.36	30.68	64.35	28.79	67.90	25.39	73.36	14.08	23.58	67.67	13.05	16.73	21.26	18.38
RAFT-3D (K)	33.23	32.68	60.54	30.54	71.57	27.94	70.51	28.43	24.22	62.19	13.20	27.96	28.64	11.82
CamLiFlow (F)	46.85	46.34	72.05	44.75	76.75	42.82	75.24	11.98	42.44	89.06	31.88	18.42	40.62	21.79
M-FUSE (K)	60.03	59.81	70.98	58.26	85.25	57.42	78.41	78.19	48.97	74.81	20.36	49.10	54.22	19.50
RAFT-3D (F)	77.57	77.43	84.67	77.14	83.63	78.25	72.80	80.42	81.66	63.87	67.58	18.42	72.27	48.80
CamLiFlow (K)	84.35	84.21	91.19	83.55	95.65	83.00	93.85	55.39	87.79	99.85	69.03	27.96	74.99	67.72

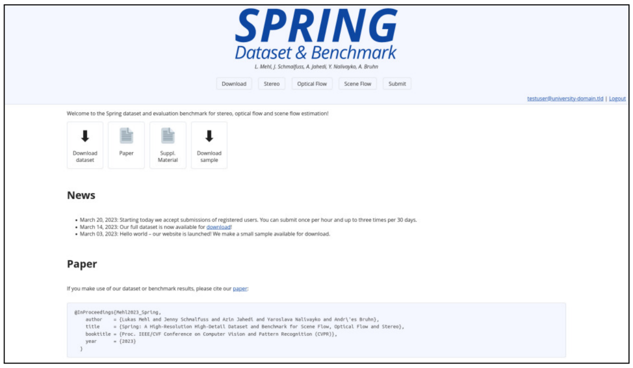

Figure 5. Spring benchmark: Start page.

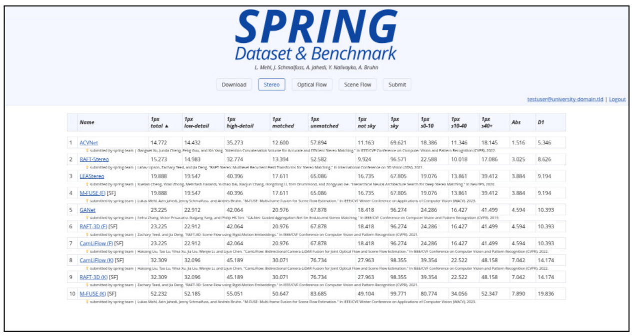

Figure 6. Spring benchmark: Stereo ranking.

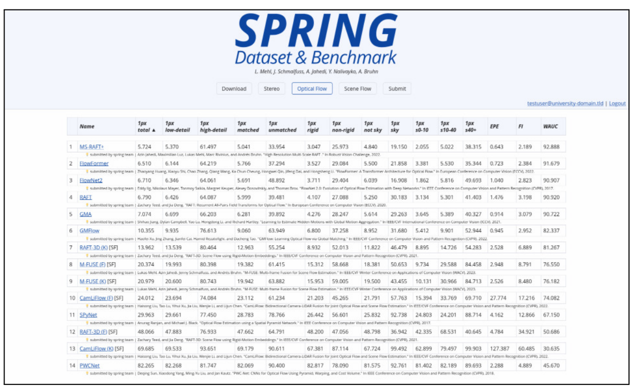

Figure 7. Spring benchmark: Optical flow ranking.

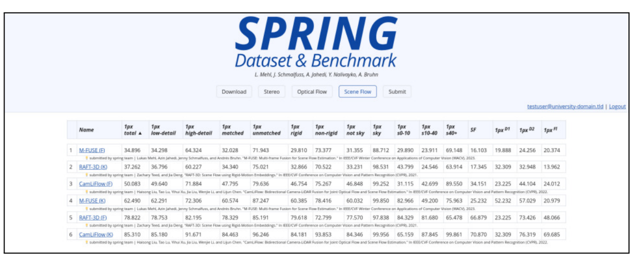

Figure 8. Spring benchmark: Scene flow ranking.

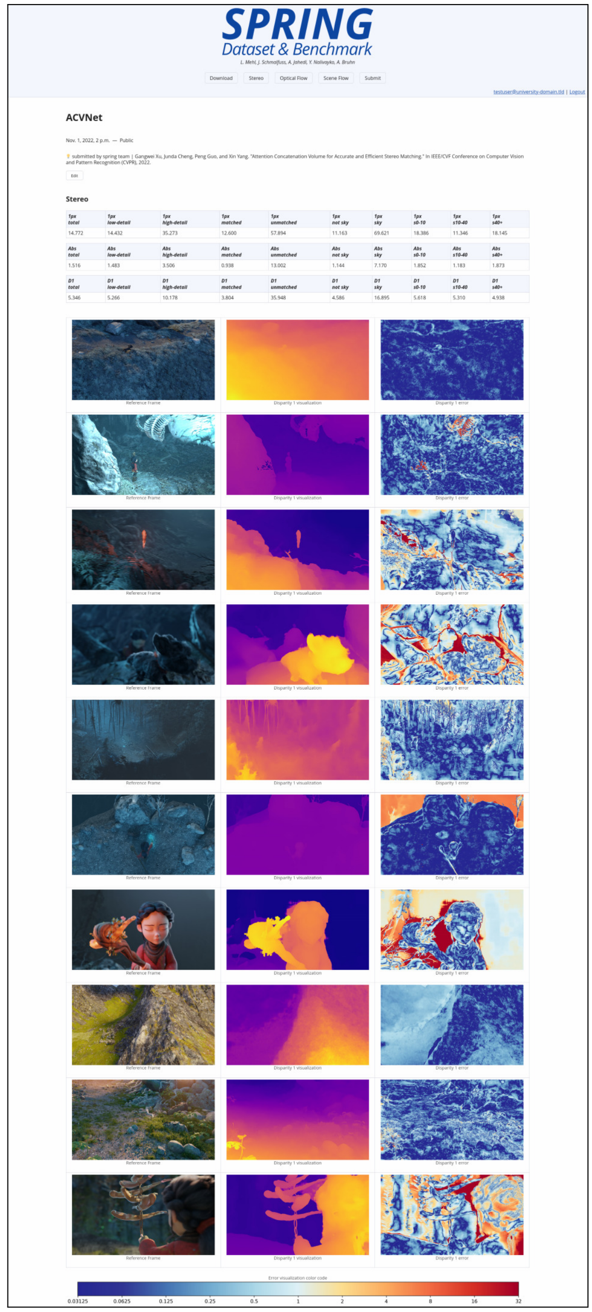

Figure 9. Spring benchmark: Example of stereo result.

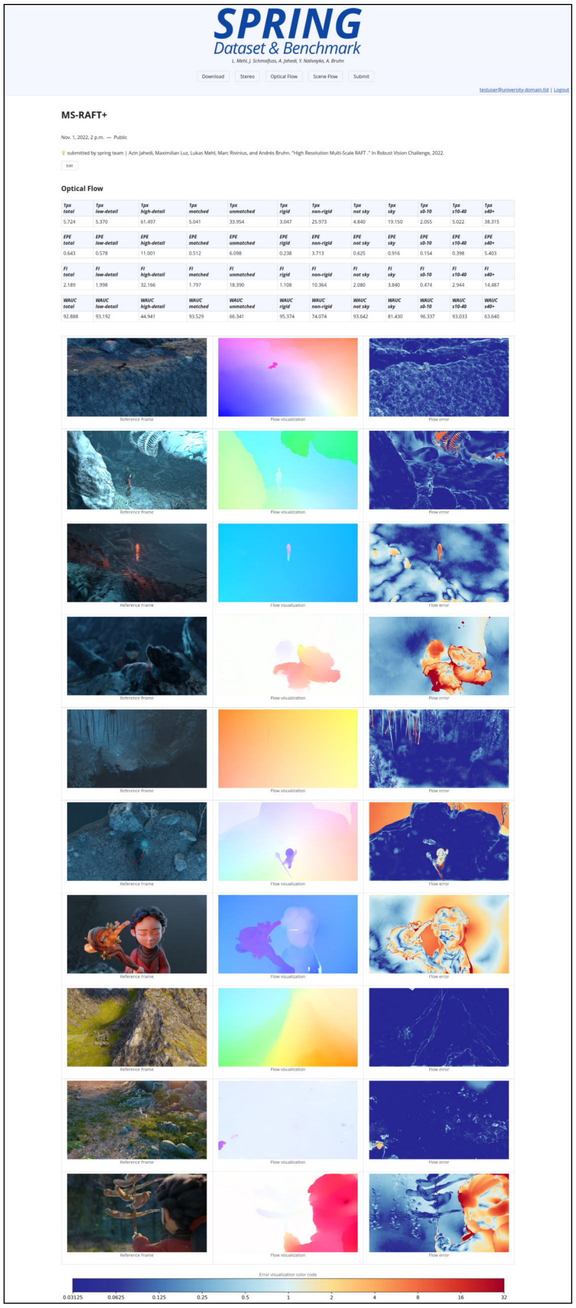

Figure 10. Spring benchmark: Example of optical flow result.

Figure 11. Spring benchmark: Example of scene flow result.

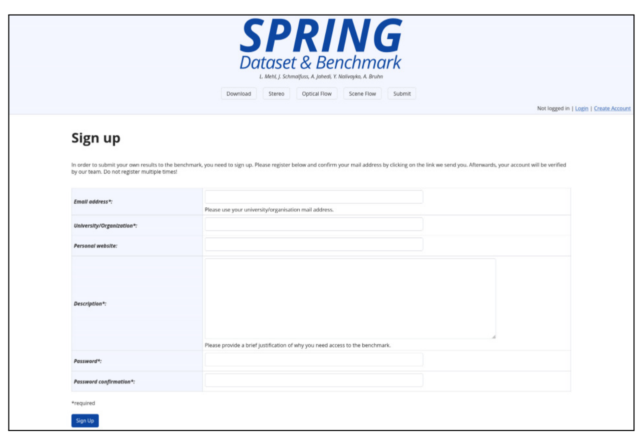

Figure 12. Spring benchmark: Registration form.