Supplementary Material for Mono-SF: Multi-View Geometry Meets Single-View Depth for Monocular Scene Flow Estimation of Dynamic Traffic Scenes

Fabian Brickwedde[1,2] Steffen Abraham[1] Rudolf Mester[3,2]
[1] Robert Bosch GmbH, Hildesheim, Germany
[2] VSI Lab, CS Dept., Goethe University, Frankfurt, Germany
[3] Norwegian Open AI Lab, CS Dept. (IDI), NTNU Trondheim, Norway
{Fabian.Brickwedde;Steffen.Abraham}@de.bosch.com

In this supplementary material, additional results and discussions of the *Mono-SF* and *ProbDepthNet* approaches presented in the paper are provided. The first section provides analyzes and results of ProbDepthNet for probabilistic single-view depth estimation and is divided into three parts. First, exemplary estimates based on images of the KITTI scene flow training set [15], which is the dataset the ProbDepthNet model is trained for, are shown. Second, a quantitative evaluation of ProbDepthNet with respect to state-of-the-art methods for single-view depth estimation is presented. Third, ProbDepthNet is tested on the Cityscapes [3] and Make3D [16] datasets to analyze the generalization capabilities. In the second section, qualitative results of the Mono-SF approach for monocular scene flow estimation are provided. The examples cover a wide range of scenarios of the KITTI scene flow training set [15] and the results of three state-of-the-art monocular baseline methods are shown as well.

1. ProbDepthNet for Probabilistic Single-View Depth Estimation

The following experiments are conducted on *ProbDepthNet* models, which are trained as described in the paper. ProbDepthNet provides pixel-wise probabilistic single-view depth estimates in the form of a mixture of Gaussians with 8 components. The results of each component are visually similar, which is why only the mean depth values μ_0 and variances $s_0, \tilde{s}_0$ of the first component are shown. The variances s_0 denote the output of DepthNet and the variances $\tilde{s}_0$ are the recalibrated ones of the additional CalibNet. The mean depth values are colored from close (red) to far (blue). The variances are colored in red shades for high values and in blue shades for low values.

1.1. Qualitative Results on KITTI

Fig. 1 shows exemplary estimates of ProbDepthNet on the KITTI scene flow training set [15]. The estimates are based on a ProbDepthNet model, which is pretrained on Cityscapes [3] and fine-tuned on the KITTI raw dataset [7] excluding the sequences that are part of the KITTI scene flow training set. The results show that the recalibrated variances $\tilde{s}_0$ are higher than the variances s_0. This recalibration technique achieves well-calibrated distributions by compensating overfitting effects as shown in the paper. Furthermore, the results support that the variances correlate with the errors of the mean depth values. Thereby, high variances are typically estimated in the following situations. First, ProbDepthNet estimates high variances correctly for challenging parts such as thin objects (e.g. the poles in Fig. 1(a,c)) or object boundaries (e.g. the object boundaries of the vehicles in Fig. 1(a,b,d,e)). The variances are lower for the object boundaries at the bottom than at the top of the vehicles. This is due to the fact that the difference in depth is larger between the vehicle and the background than between the vehicle and the ground plane. Second, ProbDepthNet is able to estimate high variances for parts that lack valuable ground truth data for training. For example, the stereo-based completion of the lidar data does not provide valuable ground

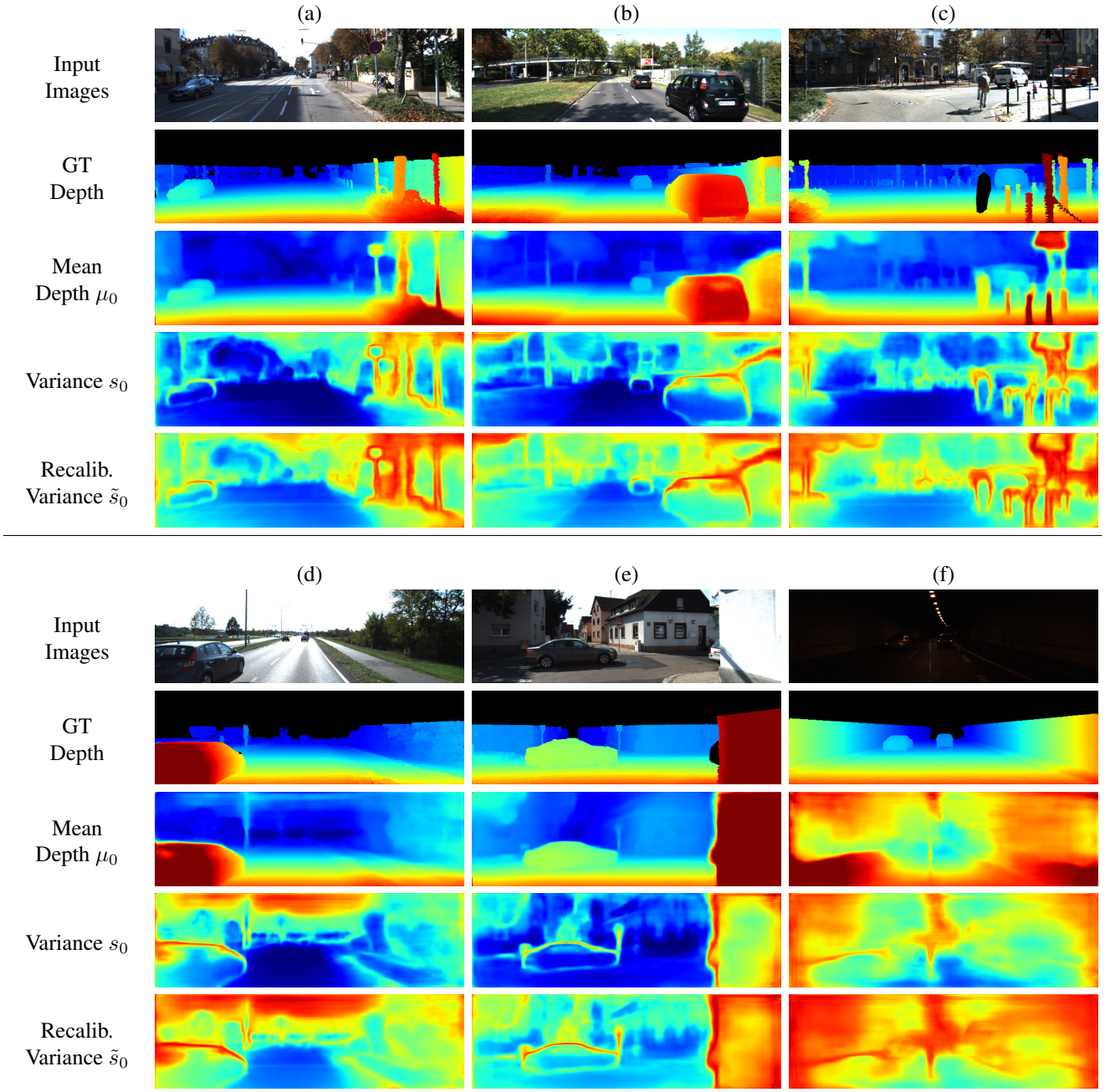

Figure 1. Exemplary estimates of ProbDepthNet on the KITTI scene flow dataset [15] in the form of the mean depth values μ_0, variances s_0 and recalibrated variances $\tilde{s}_0$ of the first component of the mixture of Gaussians. The color encodes the inverse depth from close (red) to far (blue) or the variance from high (red) to low (blue).

truth data for the low-textured sky (see Fig. 1(d)). Third, ProbDepthNet is able to identify scenarios that result in an erroneously estimated depth structure such as the dark tunnel in Fig. 1(f). In this scenario, ProbDepthNet estimates high variances correctly for almost the whole image.

1.2. Quantitative Evaluation on KITTI

Previous state-of-the-art methods for single-view depth estimation such as [4, 5, 6, 8, 11, 12] are typically designed to estimate pixel-wise depth values. In contrast to these methods, the contribution and benefit of ProbDepth-

Net are to provide well-calibrated pixel-wise depth distributions instead of single depth values. This probabilistic design is beneficial for integrating single-view depth information in the probabilistic optimization framework of Mono-SF as shown in the paper. However, to get an impression of the quality of depth information estimated by ProbDepthNet with respect to previous methods, we interpret the total means of the pixel-wise depth distributions as estimates of single depth values. Table 1 shows the quantitative evaluation of these values with respect to state-of-the-art methods for single-view depth estimation following the evaluation metric and KITTI test split proposed by Eigen et al. [4]. The results are based on a ProbDepthNet model pretrained on Cityscapes [3] and fine-tuned on the KITTI training split as used by Eigen et al. [4]. Additionally, the results of the following baseline methods are provided. Eigen et al. [4] proposed a convolutional neural network (CNN) that estimates depth values from a single image in a coarse to fine scheme. Liu et al. [12] combined a CNN with a conditional random field for single-view depth estimation. Whereas these methods are trained in a supervised manner based on lidar data, Garg et al. [6] and Godard et al. [8] (denoted as LRC) proposed self-supervised training methods using a stereo setup. Kuznietsov et al. [11] proposed a CNN trained in a semi-supervised manner by combining supervised with self-supervised training losses. The currently leading approach of the KITTI depth prediction benchmark [17] was proposed by Fu et al. [5] (denoted as DORN). They formulate the single-view depth estimation as an ordinal regression problem, which is trained in a supervised manner based on the ground truth data provided by the KITTI depth prediction benchmark [17]. The results stated in Table 1 are taken from the corresponding papers expect for the DORN [5] method. Whereas the other methods evaluate their depth estimates against the raw lidar point cloud, the DORN method was evaluated against the ground truth data provided by the KITTI depth prediction benchmark [17]. To provide a fair comparison, we take the published estimates of the DORN method [5] and evaluate these estimates using the raw lidar point cloud as ground truth. The estimates of the DORN method [5] are slightly superior to the total mean depth values interpreted as single depth estimates of ProbDepthNet. However, even though ProbDepthNet is focused on providing depth distributions, it is also comparative to state-of-the-art methods in terms of providing single depth estimates. The ablation study in the paper shows a significantly better quality of Mono-SF based on the probabilistic ProbDepthNet than based on the LRC [8] method. In terms of estimating a single depth value, the quality of ProbDepthNet is just similar or slightly superior to the LRC [8] method. This additionally supports that the main benefits of using ProbDepthNet for integrating single-view depth information in Mono-SF are due to the probabilistic design and due to providing pixel-wise depth distributions instead of single depth values.

Method	Cap	lower is better				higher is better		
		Abs Rel	Sq Rel	RMSE	RMSE_{log}	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Eigen et al. [4]	0 - 80 m	0.190	1.515	7.156	0.270	0.692	0.899	0.967
Liu et al. [12]	0 - 80 m	0.217	1.841	6.986	0.289	0.647	0.882	0.961
LRC [8]	0 - 80 m	0.114	0.898	4.935	0.206	0.861	0.949	0.976
Kuznietsov et al. [11]	0 - 80 m	0.113	0.741	4.621	0.189	0.862	0.960	0.986
DORN [5]	0 - 80 m	0.111	0.618	3.659	0.168	0.894	0.964	0.984
ProbDepthNet	0 - 80 m	0.103	0.762	4.680	0.195	0.871	0.953	0.979
Garg et al. [6]	0 - 50 m	0.169	1.080	5.104	0.273	0.740	0.904	0.962
LRC [8]	0 - 50 m	0.108	0.657	3.729	0.194	0.873	0.954	0.979
Kuznietsov et al. [11]	0 - 50 m	0.108	0.595	3.518	0.179	0.875	0.964	0.988
DORN [5]	0 - 50 m	0.108	0.535	2.884	0.162	0.902	0.966	0.985
ProbDepthNet	0 - 50 m	0.098	0.567	3.530	0.183	0.883	0.959	0.981

Abs Rel, ***Sq Rel***: *absolute and squared relative depth error;* ***RMSE***: *root mean squared depth error*

$\boldsymbol{\text{RMSE}_{log}}$: *root mean squared logarithmic depth error;* $\boldsymbol{\delta < 1.25^k}$: *frequency of estimates fulfilling a quality threshold*

Table 1. Quantitative evaluation of methods for single-view depth estimation on the KITTI dataset [7] using the test split by Eigen et al. [4].

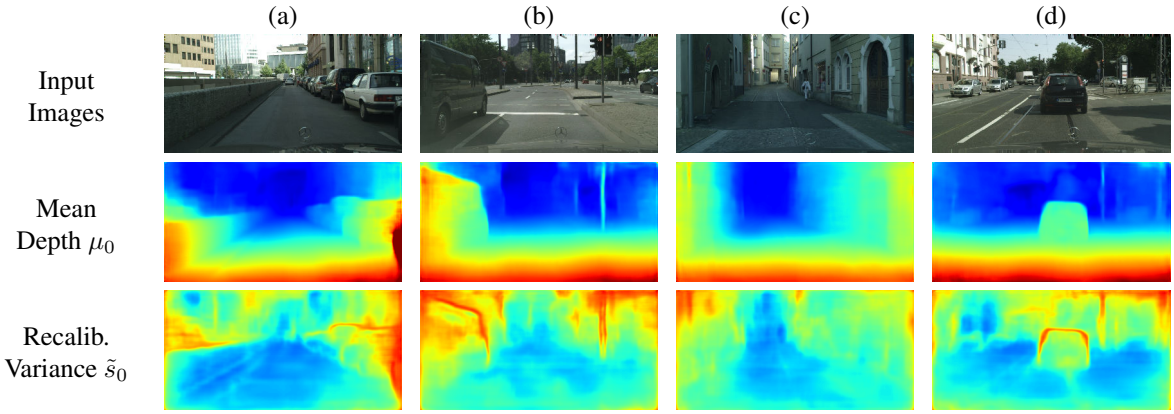

Figure 2. Generalization of ProbDepthNet (trained on KITTI [7]) on Cityscapes [3]. The figure shows the estimates based on the input image (top) in the form of the mean depth values μ_0 (middle) and recalibrated variances $\tilde{s}_0$ (bottom) of the first component of the mixture of Gaussians. The color encodes the inverse depth from close (red) to far (blue) or the variance from high (red) to low (blue).

1.3. Analysis of Generalization Capabilities on Cityscapes and Make3D

To analyze the generalization capabilities, exemplary estimates of ProbDepthNet on images of datasets different from the training data are provided. Fig. 2 shows results of a ProbDepthNet model which is trained on KITTI [15] and tested on Cityscapes [3]. Fig. 3 shows results of a ProbDepthNet model which is trained on Cityscapes and KITTI and tested on central crops of images of the Make3D [16] dataset. The KITTI and Cityscapes datasets consist of images of street scenes captured by a front-facing camera of a vehicle. Due to the similarity, a ProbDepthNet model trained on KITTI generalizes well to the Cityscapes dataset and provides reasonable estimates as shown in Fig. 2. Note, that the estimates are only correct up to a different scale due to the different focal lengths of the cameras. In contrast to these datasets, the Make3D dataset comprises images of a wide range of outdoor scenes captured by a hand-held camera. The estimates of a ProbDepthNet model trained on KITTI and Cityscapes are not only reasonable for street scenes (see Fig. 3(a,b)), but also for some different scenes (see Fig. 3(c-f)) of the Make3D dataset. There are also limits of generalization capabilities. Images that are too different from the training data such as the close-up views of buildings (see Fig. 3(d,e)) can result in erroneous estimates of both, mean depth values and variances. Note that ProbDepthNet is designed to capture the aleatoric uncertainty regarding the theory of Kendall and Gal [10]. In such scenarios an additional representation of the epistemic [10] or distributional uncertainty [13] would be needed.

2. Mono-SF for Monocular Scene Flow Estimation

In this section, additional results of monocular scene flow estimation methods are provided for the KITTI scene flow training set [15]. Our method, *Mono-SF*, presented in the corresponding paper is compared to the following three monocular baselines: First, the results of the multi-task convolutional neural network, DF-Net [18], are shown. Second, the MirrorFlow [9] method for optical flow estimation is combined with the single-view depth estimation of Godard et al. [8] to derive monocular scene flow estimates (denoted as "MirrorFlow [9] + LRC [8]"). Third, the results of the Mono-Stixel method [2], which fuses single-view depth estimates with optical flow estimates, are shown.

Fig. 4 to Fig. 11 show exemplary estimates and errors in terms of disparity at $t = 0$ (D1), disparity at $t = 1$ (D2), and optical flow (Fl). The scene flow error (SF error), which is visualized as well, is defined as the maximum of the disparity and optical flow errors. The visualizations are generated by using the KITTI scene flow evaluation tools provided by [14]. All estimates are represented at their image coordinates in the first frame at $t = 0$. The error color coding follows a logarithmic scale, where errors above 3px are colored in red shades and errors below

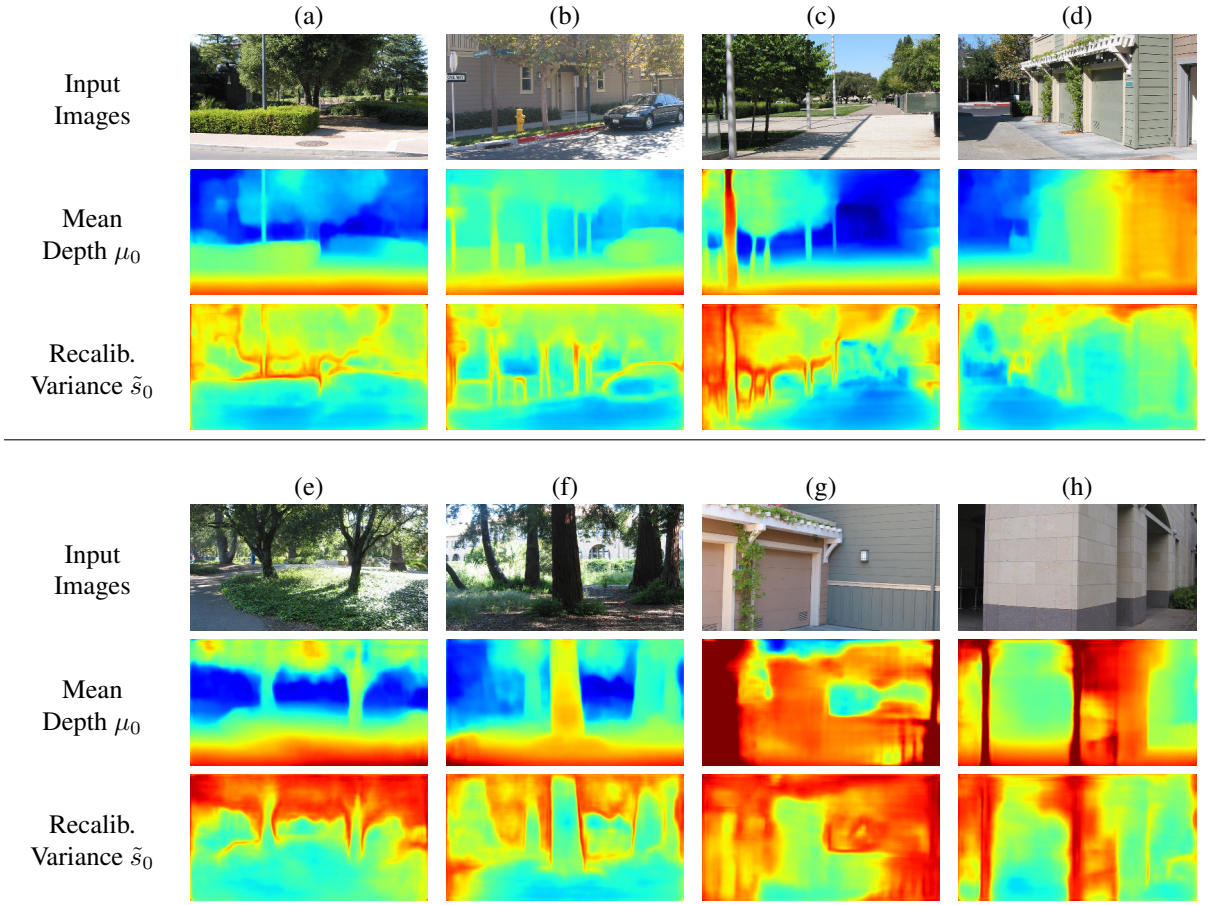

Figure 3. Generalization of ProbDepthNet (trained on Cityscapes [3] and KITTI [7]) on central crop of Make3D [16]. The figure shows the estimates based on the input image (top) in the form of the mean depth values μ_0 (middle) and recalibrated variances $\tilde{s}_0$ (bottom) of the first component of the mixture of Gaussians. The color encodes the inverse depth from close (red) to far (blue) or the variance from high (red) to low (blue).

3px are colored in blue shades.

The examples show that Mono-SF provides sharper estimates than the mentioned monocular baselines and that Mono-SF is able to reconstruct thin objects in many cases (e.g. pole and sign in Fig. 4). Even low-textured objects, that are challenging for photometric matching and optical flow estimation, are reconstructed comparatively well (see the white wall in Fig. 8). The examples show also many vehicles with several motions, like a) oncoming vehicles (see Fig. 4,6,7,10), b) preceding vehicles (see Fig. 4,9,10), c) crossing vehicles (see Fig. 5,8,11), and d) standing vehicles (see Fig. 4,7,11). Mono-SF is able to cover all these motions and to provide suitable reconstructions. Even the reconstructions of objects that do not undergo an ideal rigid body motion are visually plausible (see the pedestrian in Fig. 5).

Additionally, some challenges or limitations of the Mono-SF method are provided in these examples. The quality of Mono-SF mainly decreases in cases, where the photometric distance does not provide valuable information for the scene flow or depth estimation. This is most commonly the case if the scene points are occluded or outside the image boundary in the consecutive frame. But, this is also the case in situations without or with a quite small relative translational motion to the camera. In these cases, the depth estimation degenerates to a single-view depth estimation and is mainly defined by the ProbDepthNet estimates. Additionally, the scene smoothness priors get more dominant in these cases and especially small or thin objects might be smoothed out (see rightmost pole in Fig. 6 or traffic light in Fig. 11).

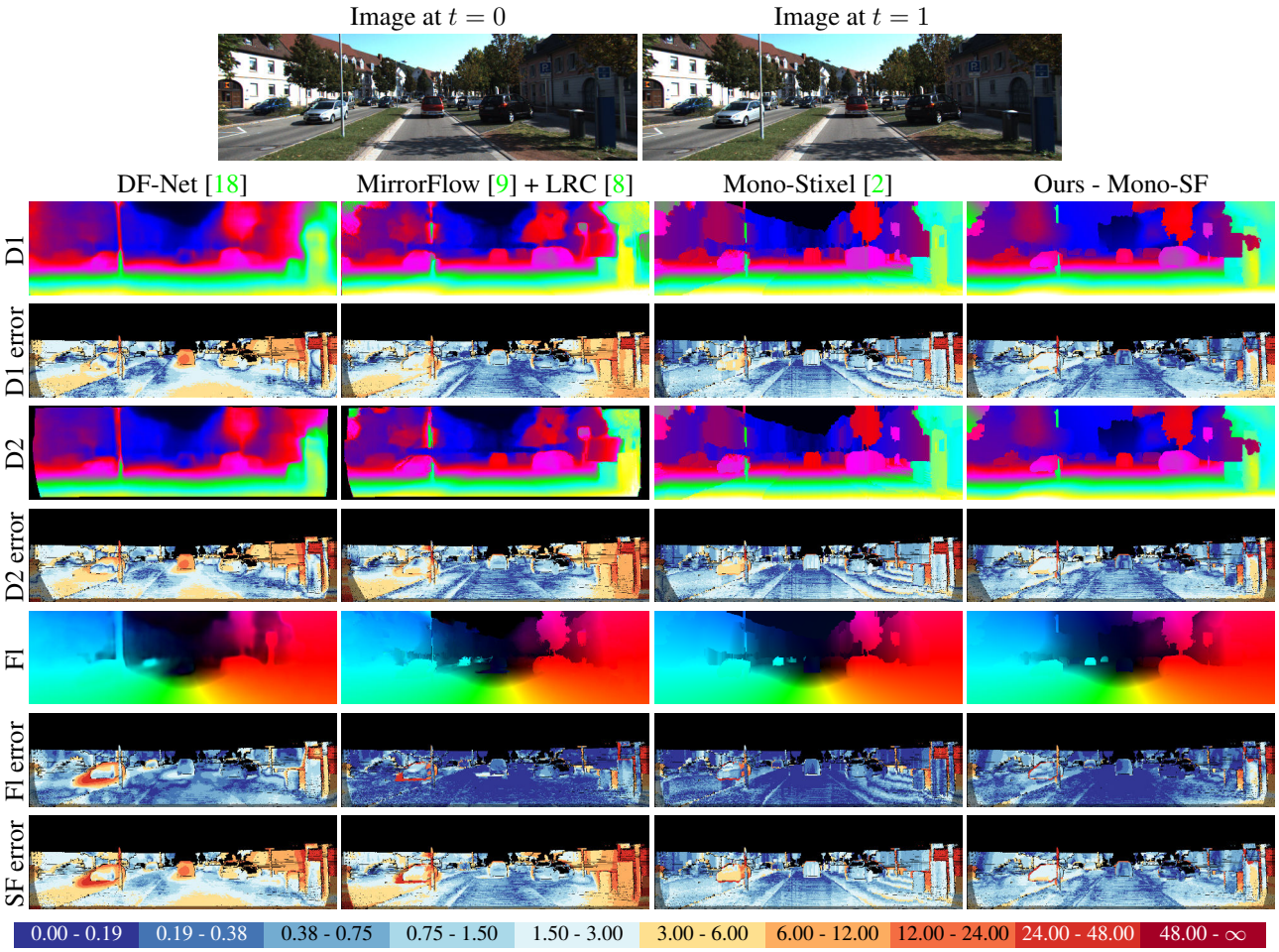

Figure 4. Exemplary results of monocular scene flow estimation methods on the KITTI scene flow training set [15]: The first three columns show the results of the monocular baselines ("DF-Net [18]", "MirrorFlow [9] + LRC [8]" and "Mono-Stixel [2]"). The fourth column corresponds to the method proposed in the corresponding paper ("Ours - Mono-SF"). From top-to-bottom, the estimates and errors of the depth at $t = 0$ (D1), of the depth at $t = 1$ (D2) and of the optical flow (Fl) are visualized. All estimates are represented at their image coordinates in the first frame. Finally, the scene flow error is shown, which is defined as the maximum of the D1, D2 and Fl errors. The depth estimates are colored from close (white/warm color) to far (blue/cool color). The optical flow is visualized following the Middlebury color coding [1]. The errors are defined as stereo disparity or optical flow endpoint errors in pixels and are colored as shown in the legend at the bottom.

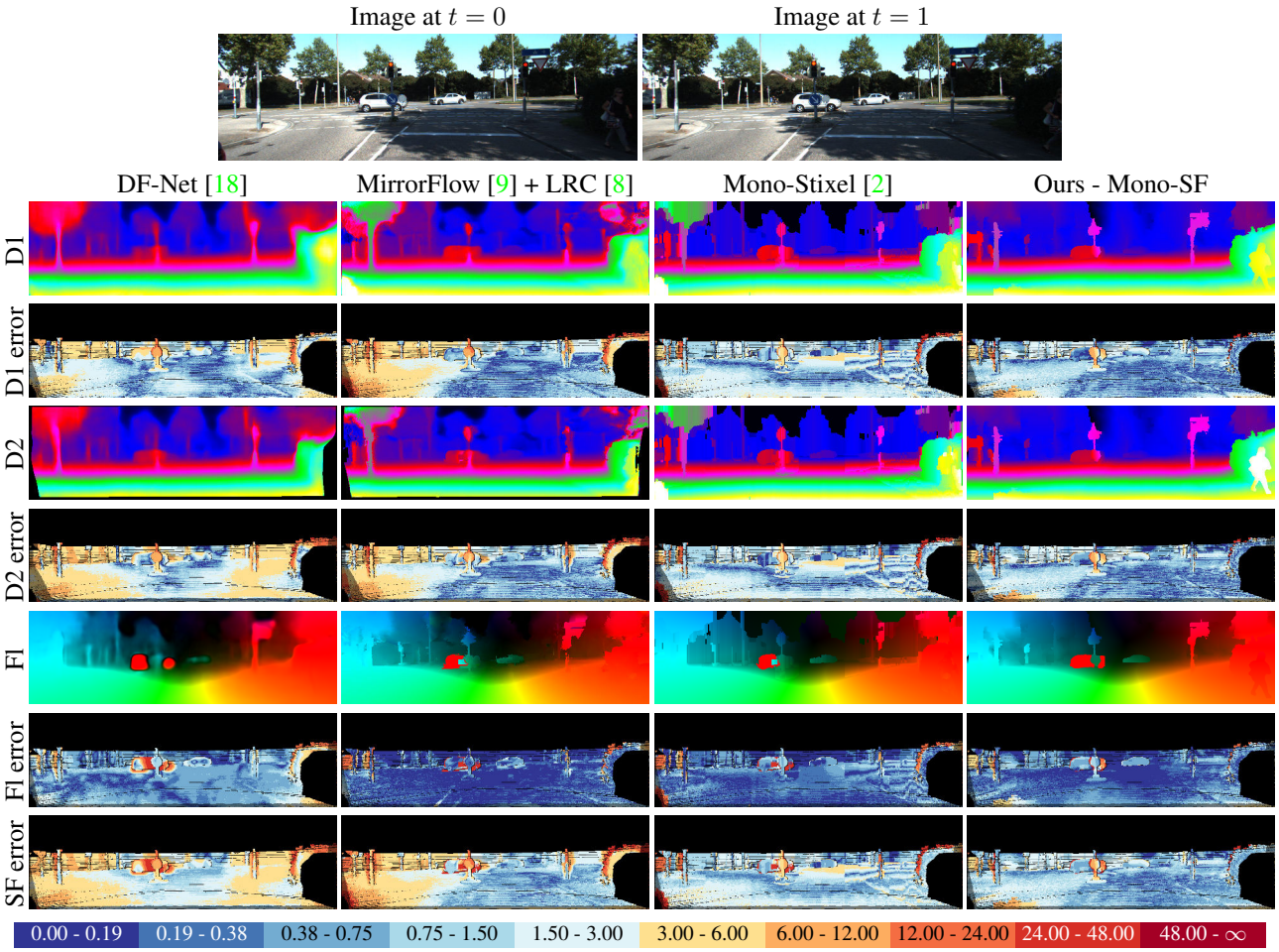

Figure 5. Exemplary results of monocular scene flow estimation methods on the KITTI scene flow training set [15]: The first three columns show the results of the monocular baselines ("DF-Net [18]", "MirrorFlow [9] + LRC [8]" and "Mono-Stixel [2]"). The fourth column corresponds to the method proposed in the corresponding paper ("Ours - Mono-SF"). From top-to-bottom, the estimates and errors of the depth at $t = 0$ (D1), of the depth at $t = 1$ (D2) and of the optical flow (Fl) are visualized. All estimates are represented at their image coordinates in the first frame. Finally, the scene flow error is shown, which is defined as the maximum of the D1, D2 and Fl errors. The depth estimates are colored from close (white/warm color) to far (blue/cool color). The optical flow is visualized following the Middlebury color coding [1]. The errors are defined as stereo disparity or optical flow endpoint errors in pixels and are colored as shown in the legend at the bottom.

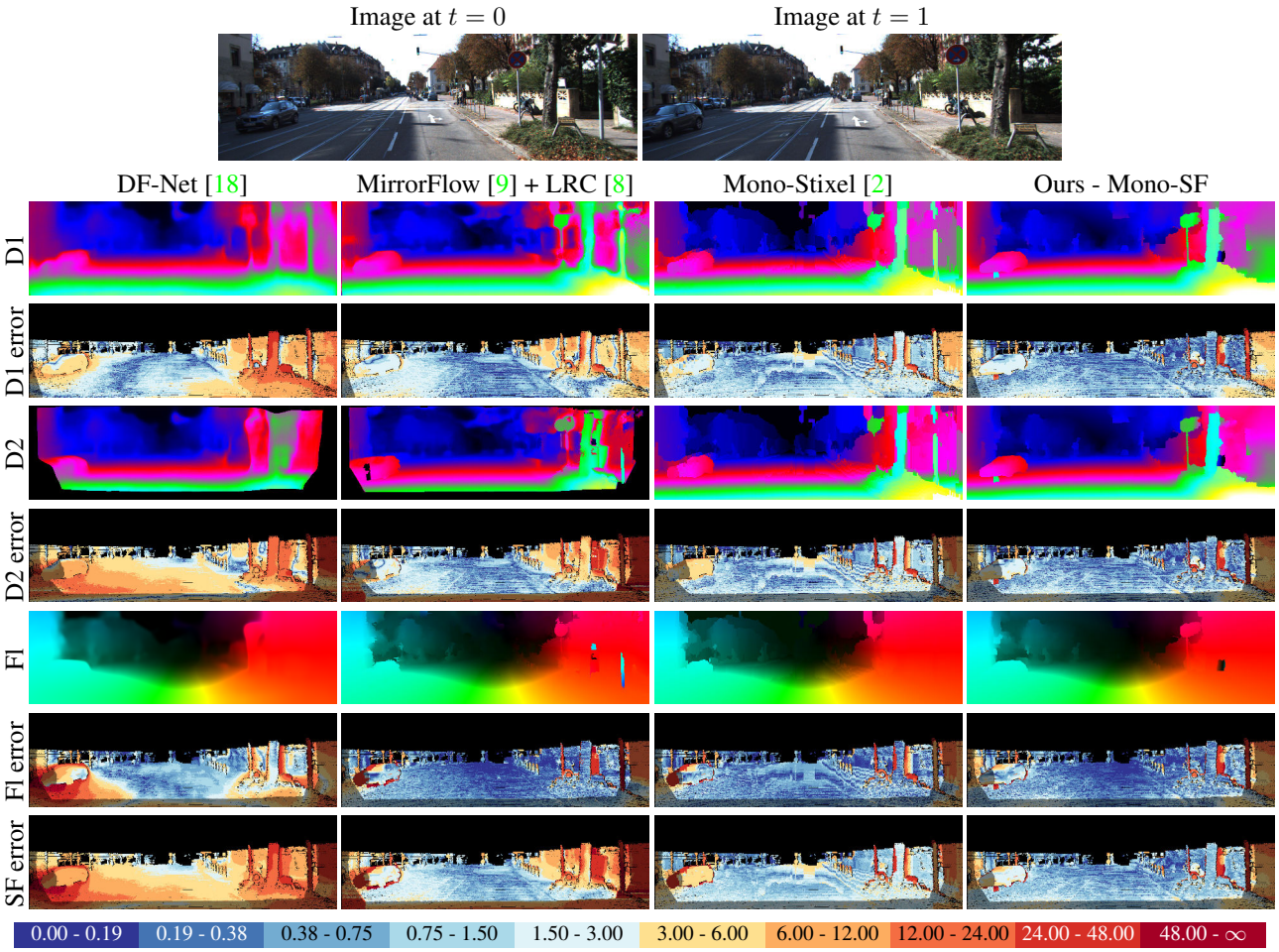

Figure 6. Exemplary results of monocular scene flow estimation methods on the KITTI scene flow training set [15]: The first three columns show the results of the monocular baselines ("DF-Net [18]", "MirrorFlow [9] + LRC [8]" and "Mono-Stixel [2]"). The fourth column corresponds to the method proposed in the corresponding paper ("Ours - Mono-SF"). From top-to-bottom, the estimates and errors of the depth at $t = 0$ (D1), of the depth at $t = 1$ (D2) and of the optical flow (Fl) are visualized. All estimates are represented at their image coordinates in the first frame. Finally, the scene flow error is shown, which is defined as the maximum of the D1, D2 and Fl errors. The depth estimates are colored from close (white/warm color) to far (blue/cool color). The optical flow is visualized following the Middlebury color coding [1]. The errors are defined as stereo disparity or optical flow endpoint errors in pixels and are colored as shown in the legend at the bottom.

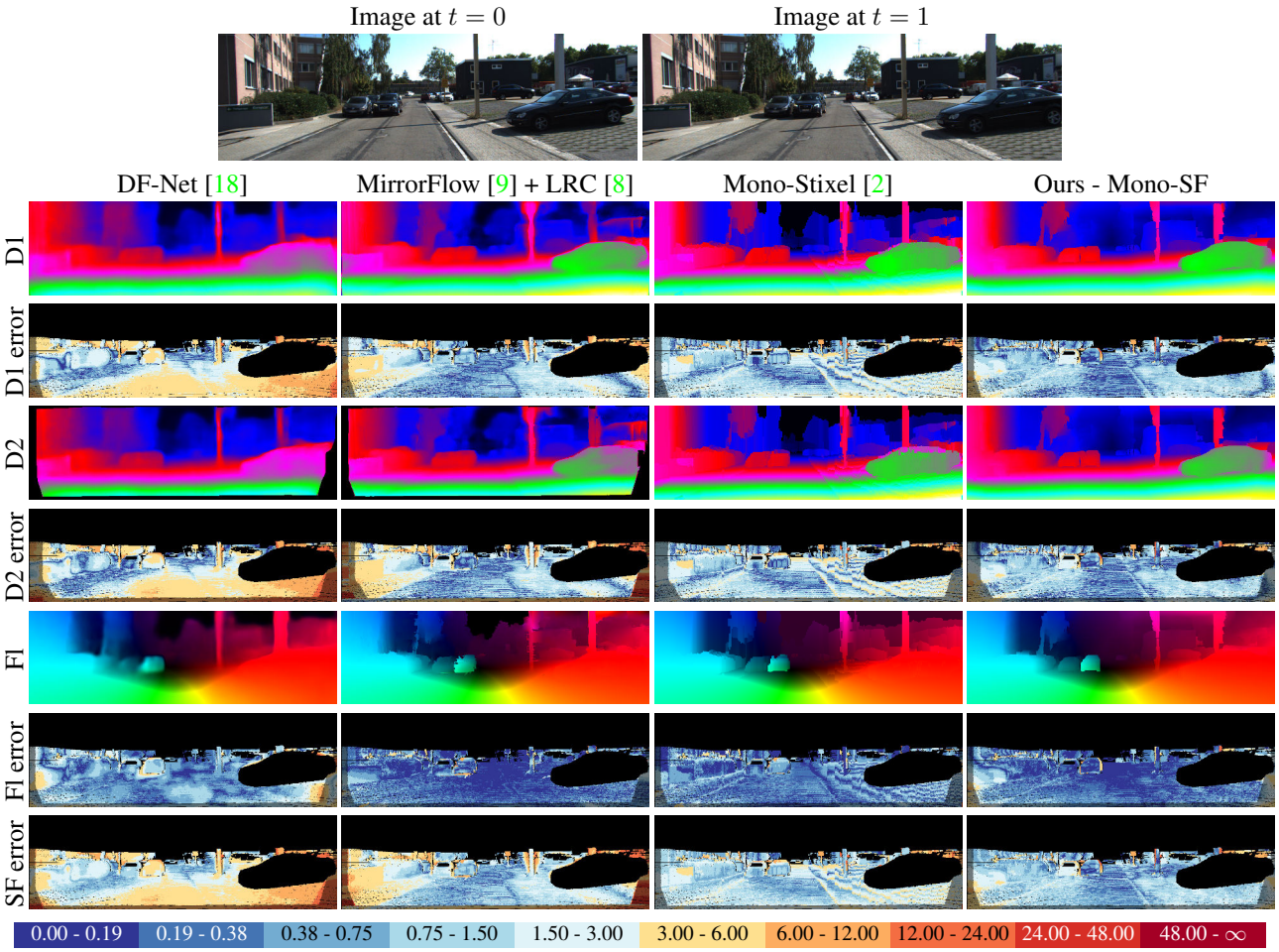

Figure 7. Exemplary results of monocular scene flow estimation methods on the KITTI scene flow training set [15]: The first three columns show the results of the monocular baselines ("DF-Net [18]", "MirrorFlow [9] + LRC [8]" and "Mono-Stixel [2]"). The fourth column corresponds to the method proposed in the corresponding paper ("Ours - Mono-SF"). From top-to-bottom, the estimates and errors of the depth at $t = 0$ (D1), of the depth at $t = 1$ (D2) and of the optical flow (Fl) are visualized. All estimates are represented at their image coordinates in the first frame. Finally, the scene flow error is shown, which is defined as the maximum of the D1, D2 and Fl errors. The depth estimates are colored from close (white/warm color) to far (blue/cool color). The optical flow is visualized following the Middlebury color coding [1]. The errors are defined as stereo disparity or optical flow endpoint errors in pixels and are colored as shown in the legend at the bottom.

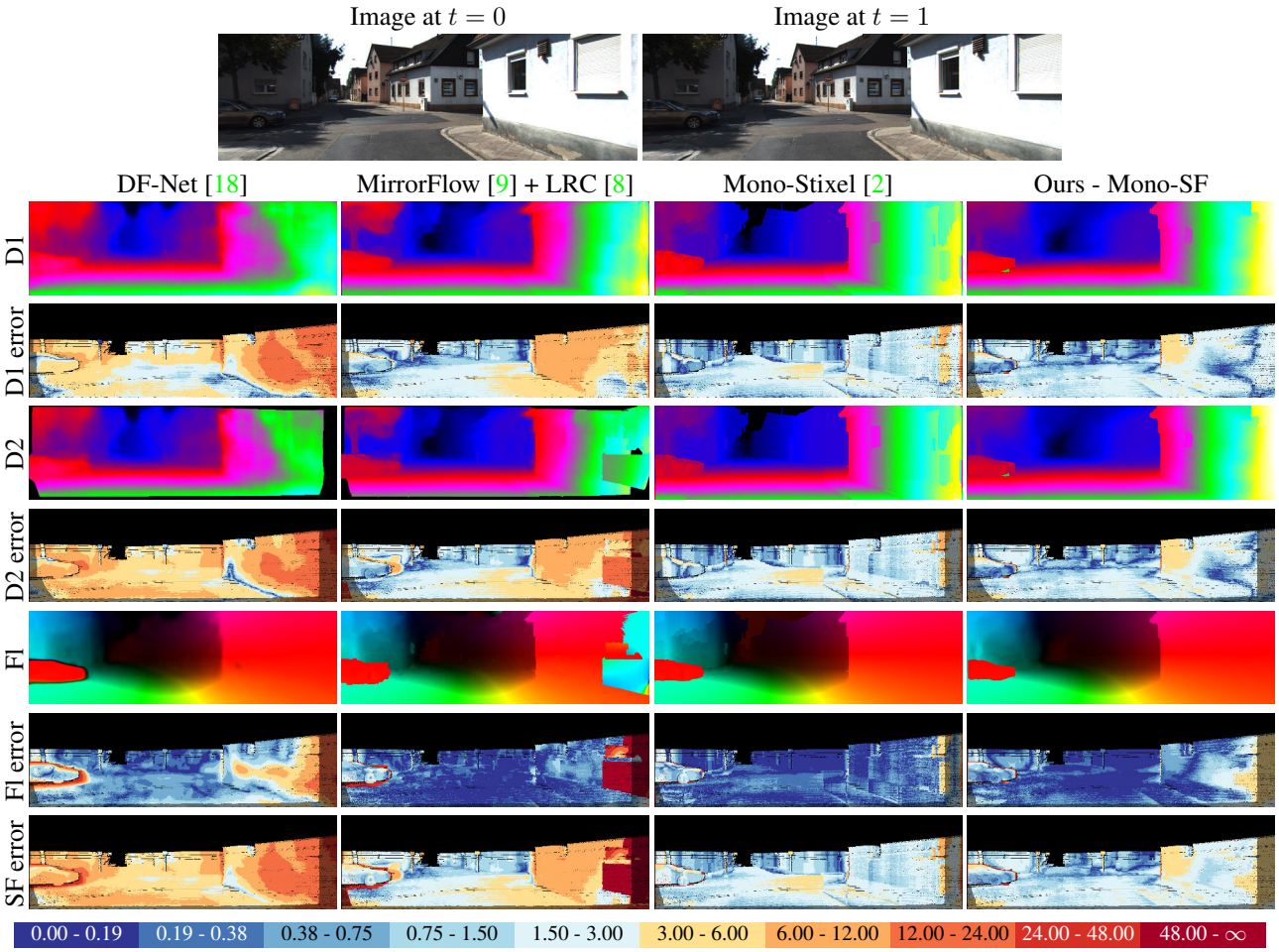

Figure 8. Exemplary results of monocular scene flow estimation methods on the KITTI scene flow training set [15]: The first three columns show the results of the monocular baselines ("DF-Net [18]", "MirrorFlow [9] + LRC [8]" and "Mono-Stixel [2]"). The fourth column corresponds to the method proposed in the corresponding paper ("Ours - Mono-SF"). From top-to-bottom, the estimates and errors of the depth at $t = 0$ (D1), of the depth at $t = 1$ (D2) and of the optical flow (Fl) are visualized. All estimates are represented at their image coordinates in the first frame. Finally, the scene flow error is shown, which is defined as the maximum of the D1, D2 and Fl errors. The depth estimates are colored from close (white/warm color) to far (blue/cool color). The optical flow is visualized following the Middlebury color coding [1]. The errors are defined as stereo disparity or optical flow endpoint errors in pixels and are colored as shown in the legend at the bottom.

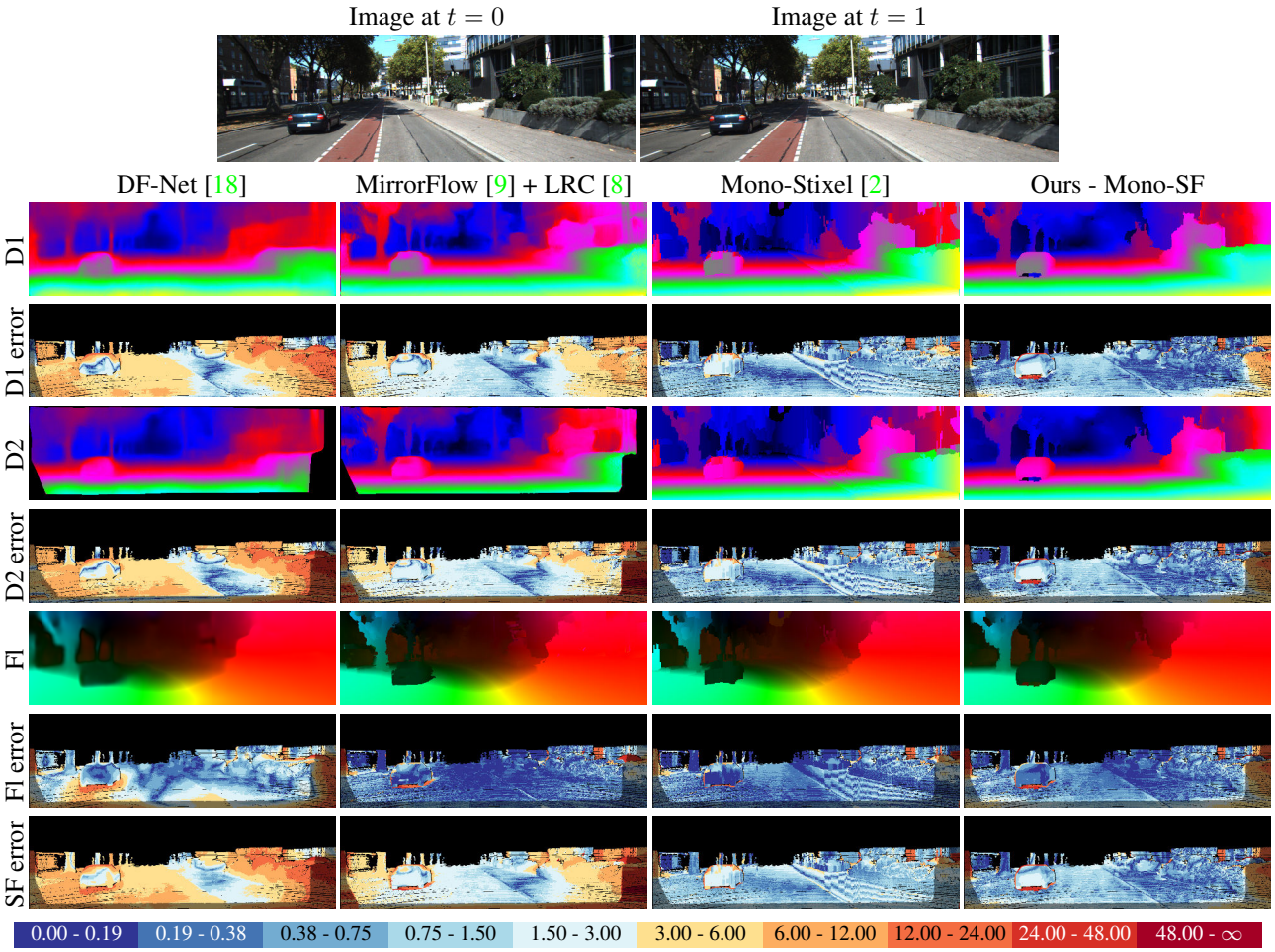

Figure 9. Exemplary results of monocular scene flow estimation methods on the KITTI scene flow training set [15]: The first three columns show the results of the monocular baselines ("DF-Net [18]", "MirrorFlow [9] + LRC [8]" and "Mono-Stixel [2]"). The fourth column corresponds to the method proposed in the corresponding paper ("Ours - Mono-SF"). From top-to-bottom, the estimates and errors of the depth at $t = 0$ (D1), of the depth at $t = 1$ (D2) and of the optical flow (Fl) are visualized. All estimates are represented at their image coordinates in the first frame. Finally, the scene flow error is shown, which is defined as the maximum of the D1, D2 and Fl errors. The depth estimates are colored from close (white/warm color) to far (blue/cool color). The optical flow is visualized following the Middlebury color coding [1]. The errors are defined as stereo disparity or optical flow endpoint errors in pixels and are colored as shown in the legend at the bottom.

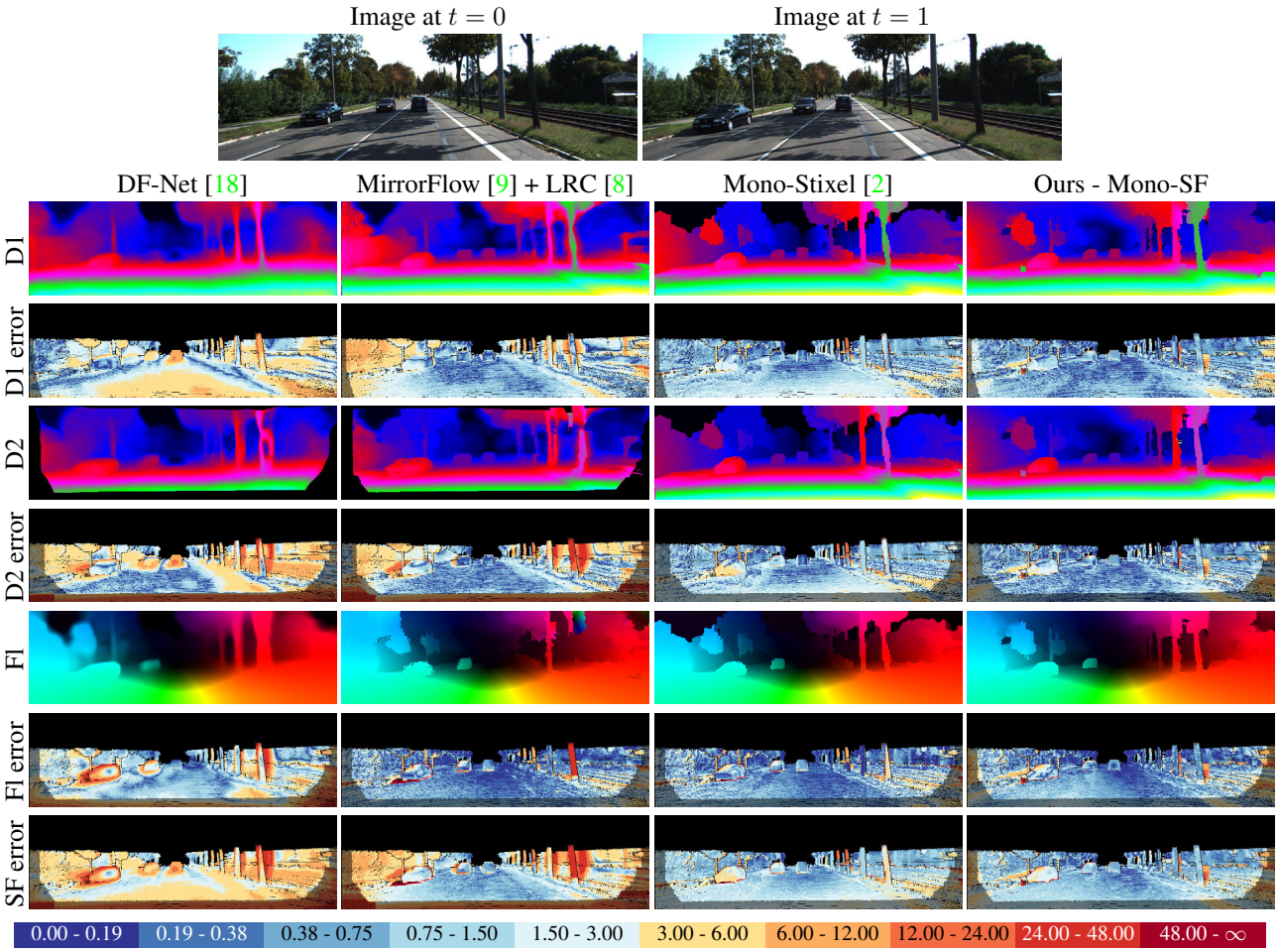

Figure 10. Exemplary results of monocular scene flow estimation methods on the KITTI scene flow training set [15]: The first three columns show the results of the monocular baselines ("DF-Net [18]", "MirrorFlow [9] + LRC [8]" and "Mono-Stixel [2]"). The fourth column corresponds to the method proposed in the corresponding paper ("Ours - Mono-SF"). From top-to-bottom, the estimates and errors of the depth at $t = 0$ (D1), of the depth at $t = 1$ (D2) and of the optical flow (Fl) are visualized. All estimates are represented at their image coordinates in the first frame. Finally, the scene flow error is shown, which is defined as the maximum of the D1, D2 and Fl errors. The depth estimates are colored from close (white/warm color) to far (blue/cool color). The optical flow is visualized following the Middlebury color coding [1]. The errors are defined as stereo disparity or optical flow endpoint errors in pixels and are colored as shown in the legend at the bottom.

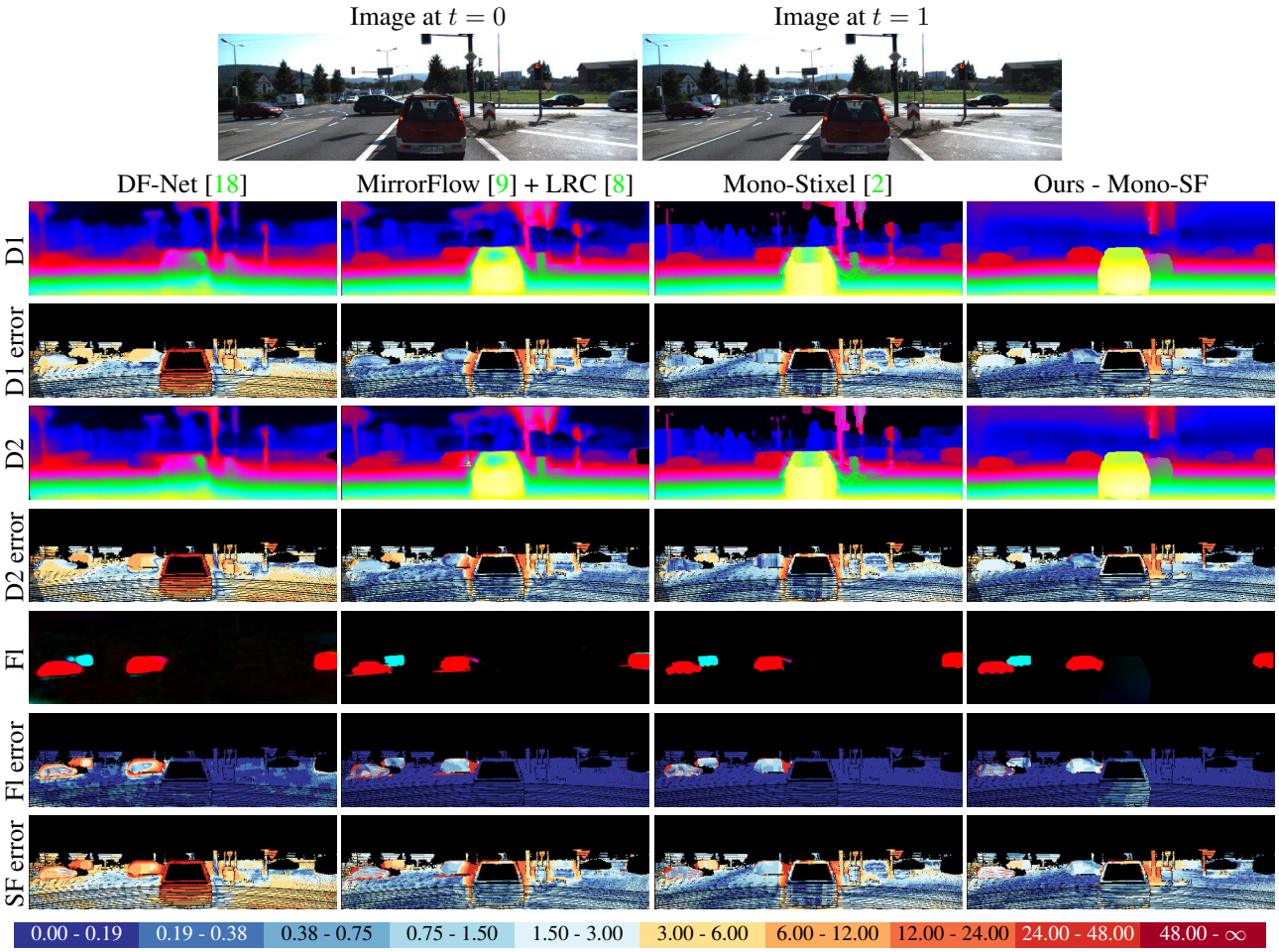

Figure 11. Exemplary results of monocular scene flow estimation methods on the KITTI scene flow training set [15]: The first three columns show the results of the monocular baselines ("DF-Net [18]", "MirrorFlow [9] + LRC [8]" and "Mono-Stixel [2]"). The fourth column corresponds to the method proposed in the corresponding paper ("Ours - Mono-SF"). From top-to-bottom, the estimates and errors of the depth at $t = 0$ (D1), of the depth at $t = 1$ (D2) and of the optical flow (Fl) are visualized. All estimates are represented at their image coordinates in the first frame. Finally, the scene flow error is shown, which is defined as the maximum of the D1, D2 and Fl errors. The depth estimates are colored from close (white/warm color) to far (blue/cool color). The optical flow is visualized following the Middlebury color coding [1]. The errors are defined as stereo disparity or optical flow endpoint errors in pixels and are colored as shown in the legend at the bottom.

References

[1] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, 92(1):1–31, 2011. 6, 7, 8, 9, 10, 11, 12, 13

[2] Fabian Brickwedde, Steffen Abraham, and Rudolf Mester. Exploiting Single Image Depth Prediction for Mono-Stixel Estimation. In *Proc. of European Conference of Computer Vision Workshops (ECCV Workshops)*. IEEE, 2018. 4, 6, 7, 8, 9, 10, 11, 12, 13

[3] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016. 1, 3, 4, 5

[4] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Proc. of Advances in neural information processing systems (NeurIPS)*, pages 2366–2374, 2014. 2, 3

[5] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep Ordinal Regression Network for Monocular Depth Estimation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2002–2011, 2018. 2, 3

[6] Ravi Garg, Gustavo Carneiro, and Ian Reid. Unsupervised CNN for single view depth estimation: Geometry to the rescue. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 740–756. Springer, 2016. 2, 3

[7] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets Robotics: The KITTI Dataset. *International Journal of Robotics Research (IJRR)*, 2013. 1, 3, 4, 5

[8] Clement Godard, Oisin Mac Aodha, and Gabriel J. Brostow. Unsupervised Monocular Depth Estimation With Left-Right Consistency. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 13

[9] Junhwa Hur and Stefan Roth. MirrorFlow: Exploiting symmetries in joint optical flow and occlusion estimation. In *Proc. of International Conference on Computer Vision (ICCV)*, 2017. 4, 6, 7, 8, 9, 10, 11, 12, 13

[10] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Proc. of Advances in neural information processing systems (NeurIPS)*, pages 5574–5584, 2017. 4

[11] Yevhen Kuznietsov, Jörg Stückler, and Bastian Leibe. Semi-supervised deep learning for monocular depth map prediction. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6647–6655, 2017. 2, 3

[12] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(10):2024–2039, 2016. 2, 3

[13] Andrey Malinin and Mark Gales. Predictive uncertainty estimation via prior networks. In *Proc. of Advances in Neural Information Processing Systems (NeurIPS)*, pages 7047–7058, 2018. 4

[14] Moritz Menze and Andreas Geiger. Object Scene Flow for Autonomous Vehicles. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 4

[15] Moritz Menze, Christian Heipke, and Andreas Geiger. Object Scene Flow. *ISPRS Journal of Photogrammetry and Remote Sensing*, 140:60 – 76, 2018. Geospatial Computer Vision. 1, 2, 4, 6, 7, 8, 9, 10, 11, 12, 13

[16] Ashutosh Saxena, Min Sun, and Andrew Y Ng. Make3D: Learning 3D Scene Structure from a Single Still Image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):824–840, May 2009. 1, 4, 5

[17] Jonas Uhrig, N Schneider, L Schneider, U Franke, Thomas Brox, and A Geiger. Sparsity Invariant CNNs. In *Proc. of IEEE International Conference on 3D Vision (3DV)*, 2017. 3

[18] Yuliang Zou, Zelun Luo, and Jia-Bin Huang. DF-Net: Unsupervised Joint Learning of Depth and Flow using Cross-Network Consistency. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 36–53, 2018. 4, 6, 7, 8, 9, 10, 11, 12, 13