Supplementary Material for Bounding Boxes, Segmentations and Object Coordinates: How Important is Recognition for 3D Scene Flow Estimation in Autonomous Driving Scenarios?

Aseem Behl[2,*] Omid Hosseini Jafari[1,*] Siva Karthik Mustikovela[1,*]
Hassan Abu Alhaija[1] Carsten Rother[1] Andreas Geiger[2,3]

[1]Computer Vision Lab, TU Dresden
[2]Autonomous Vision Group, MPI for Intelligent Systems Tübingen
[3]Computer Vision and Geometry Group, ETH Zürich

Abstract

This supplementary document provides additional analysis, visualizations and details. We start by providing a qualitative analysis of the impact of different levels of recognition granularity for foreground objects on scene flow estimation. Next, we analyze why 3D object coordinate cues fail to improve 3D scene flow performance beyond 2D segmentation masks. We further provide additional qualitative comparisons of our best performing method (ISF-SegMask) with other state-of-the-art methods on the KITTI 2015 scene flow test set. Next, we describe the CNN architecture that we have used for predicting 3D object coordinates. Finally, we compare our 3D object coordinate predictions with existing state-of-the-art approaches using random forests, demonstrating the quality of our predicted 3D object coordinates.

1. Impact of Recognition Granularity on 3D Scene Flow Estimation

In this section, we provide a qualitative analysis of the impact of different levels of recognition granularity for estimating the 3D scene flow of dynamic (i.e., foreground) objects. In particular, Fig. 1-28 compare the results of OSF [4] (no recognition input), *ISF-BBox* (2D bounding boxes as recognition input), *ISF-SegMask* (2D instance segmentations as recognition input) on our validation set (which is a subset of the KITTI 2015 training set). As 3D object coordinates do not increase performance beyond 2D instance segmentations (and hence yield the same estimates, i.e., the weight of the object coordinate terms are zero after optimization), we do not visualize the result of our model using 3D object coordinates. Instead, we refer the reader to the next section for a detailed analysis why 3D object coordinates do not provide any further gains.

For each scene and recognition granularity, Fig. 1-28 show the left input image at frame $t = 0$ and frame $t = 1$ along with the detected 2D bounding boxes, the 2D instance segmentation mask estimated by the network, the error images corresponding to the estimated disparity in the first frame (D1), the optical flow errors and the scene flow errors. Following [4], we use a logarithmic color coding where red shades represent errors above 3 pixels / 5% relative error and blue shades denote errors below 3 pixels / 5% relative error. Our results indicate that recognition (both 2D bounding box and 2D instance segmentation) provide large improvements for optical flow estimation (and in turn scene flow estimation) on foreground parts of the scene (note that we do not tackle the recognition of static background objects in this paper which are estimated relatively well without such priors). Furthermore, we note that instance segmentations as input improve performance in particular at the boundary of objects. We attribute this effect to the more fine grained nature of the 2D segmentation input compared to using rough 2D bounding boxes input.

[*] Joint first authors with equal contribution.

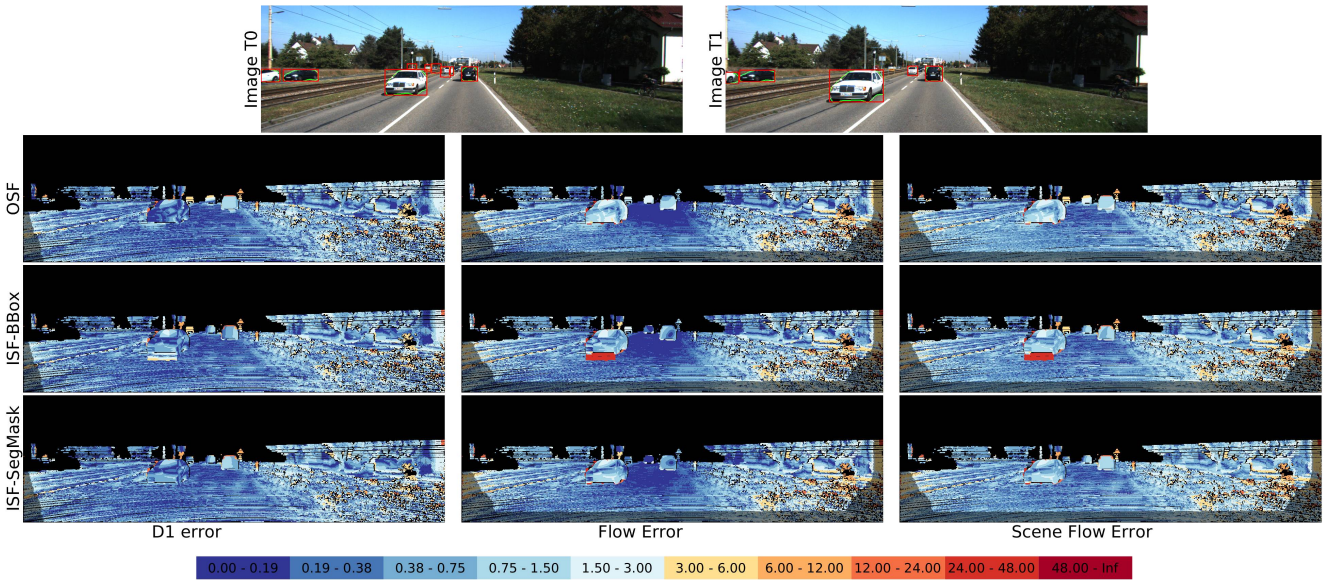

Figure 1: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

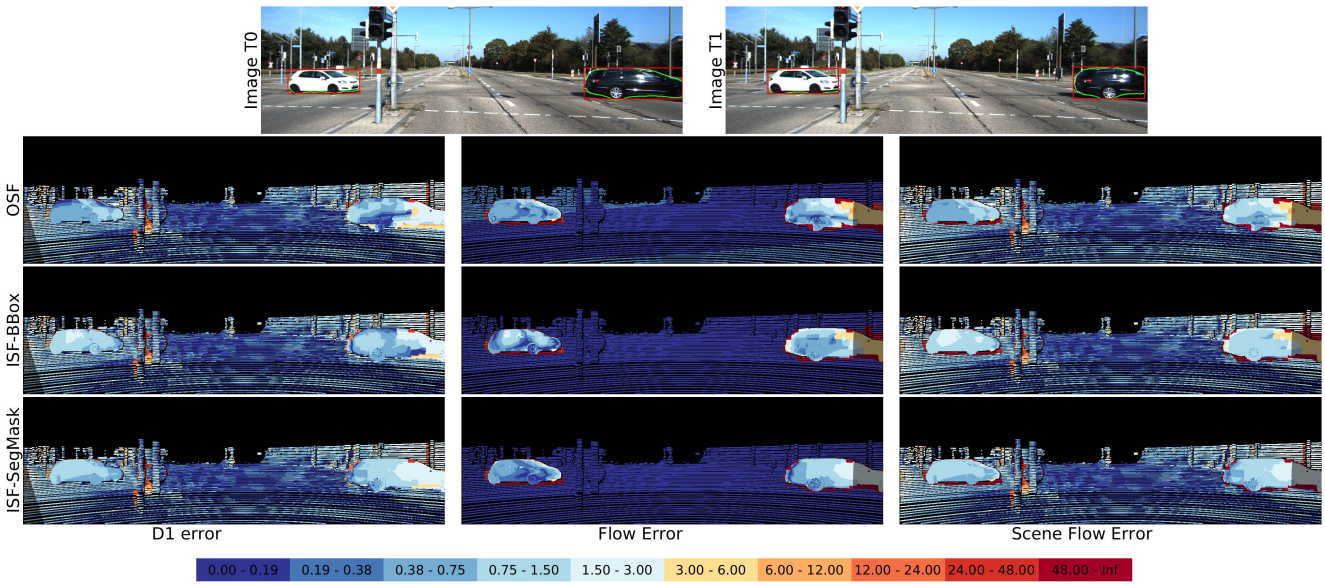

Figure 2: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

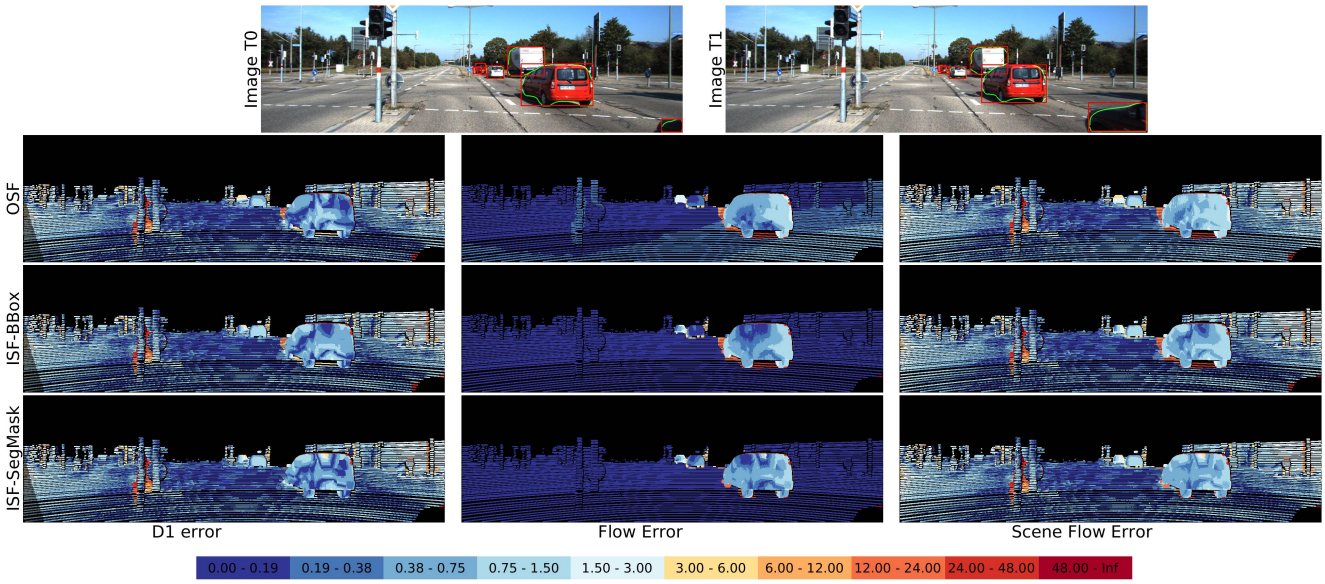

Figure 3: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

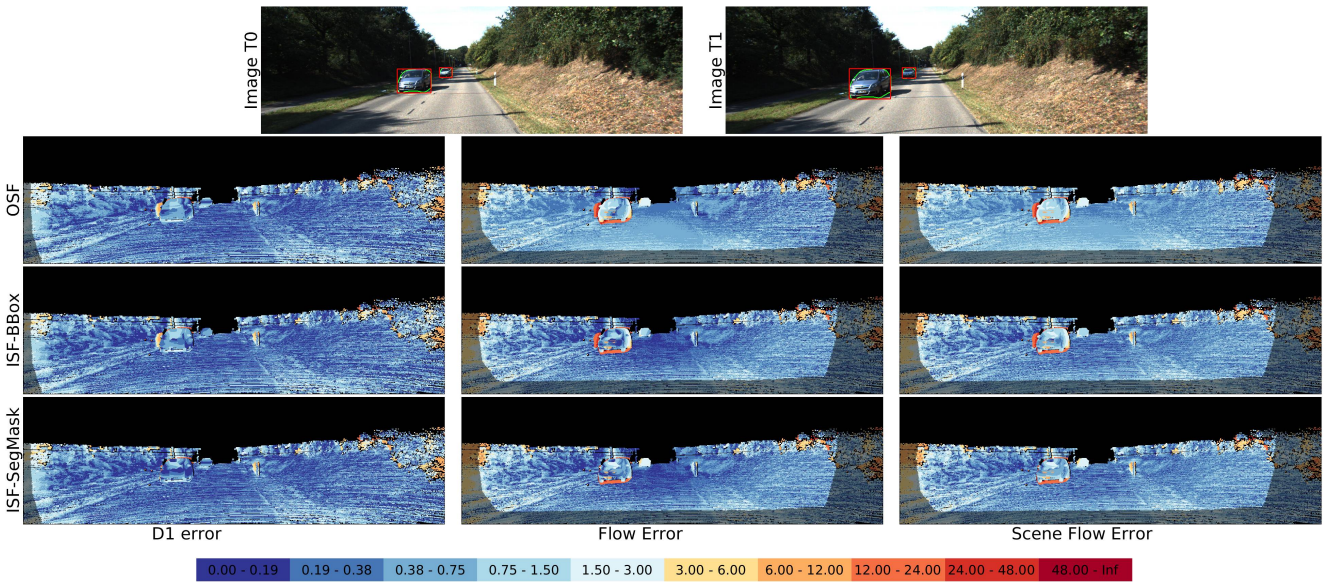

Figure 4: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

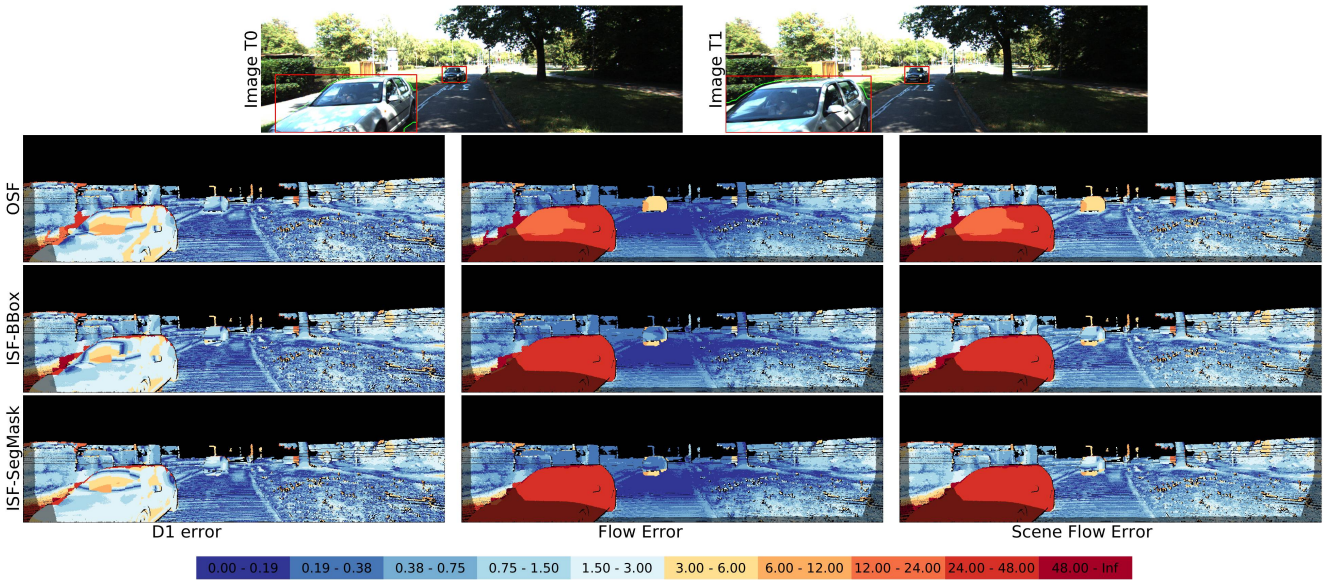

Figure 5: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

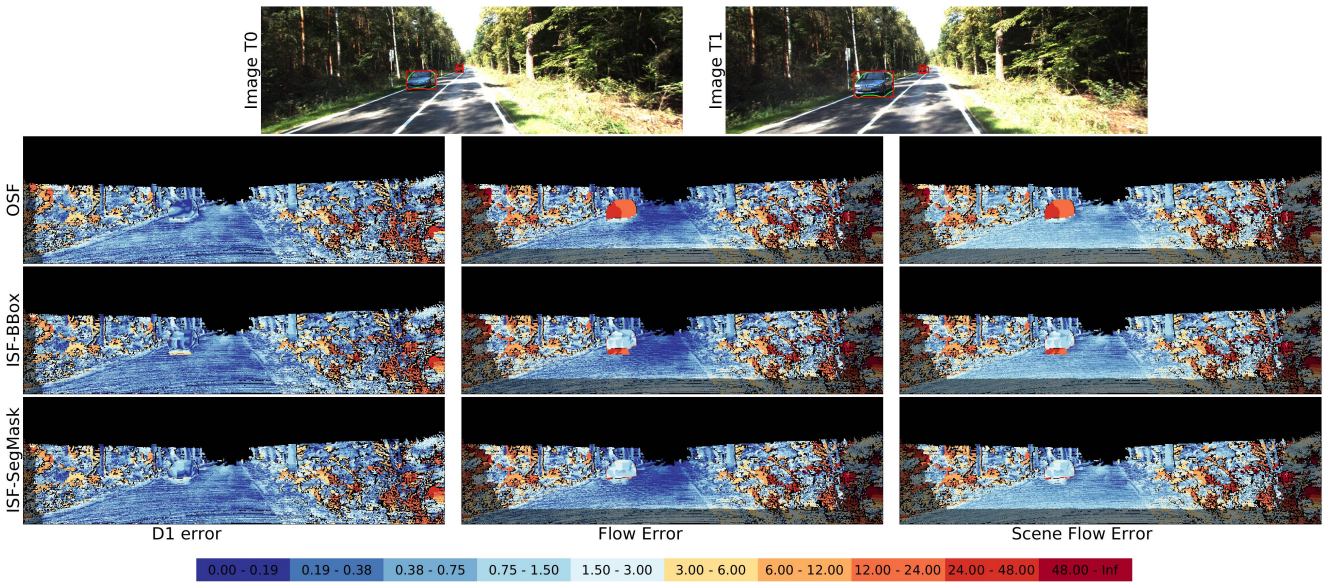

Figure 6: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

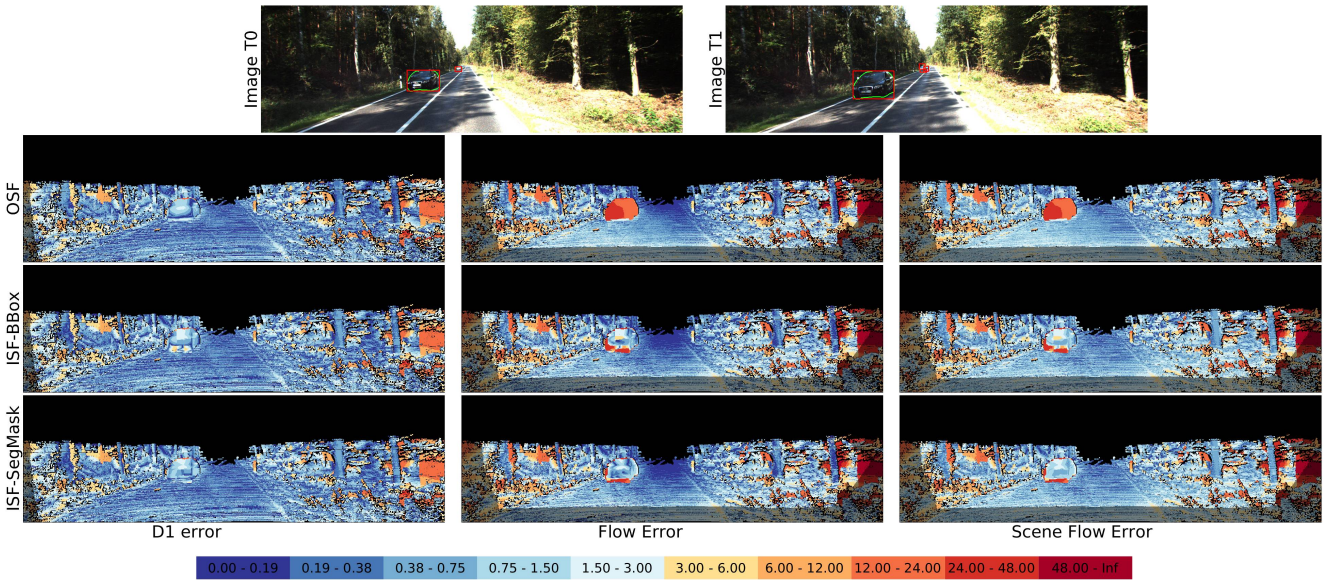

Figure 7: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

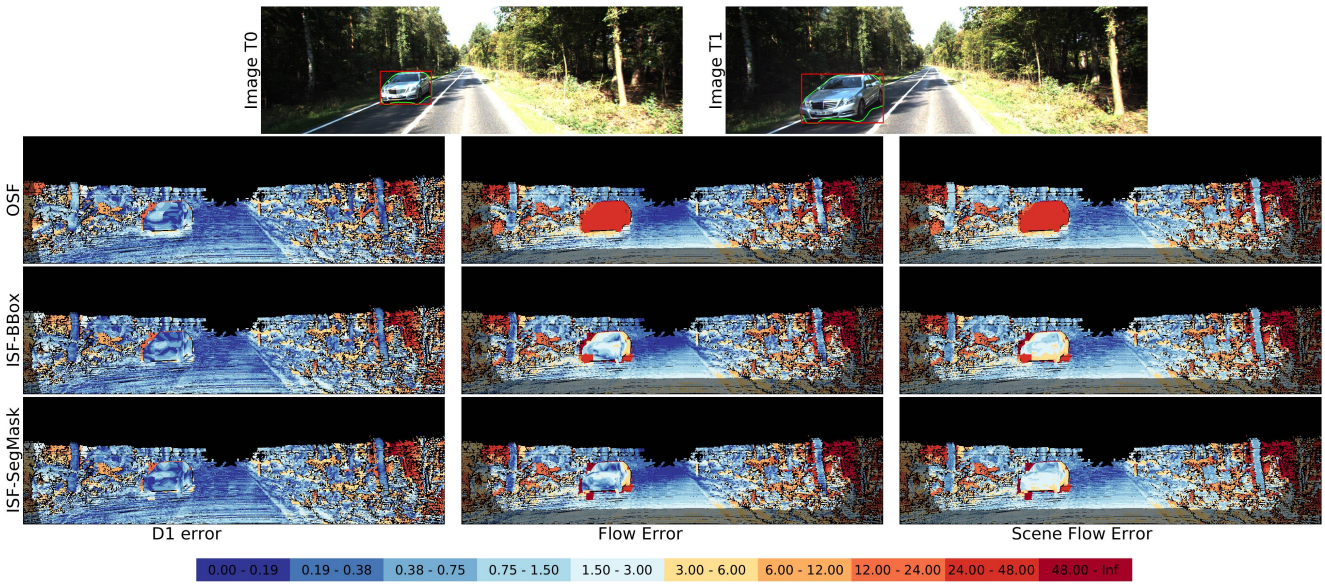

Figure 8: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

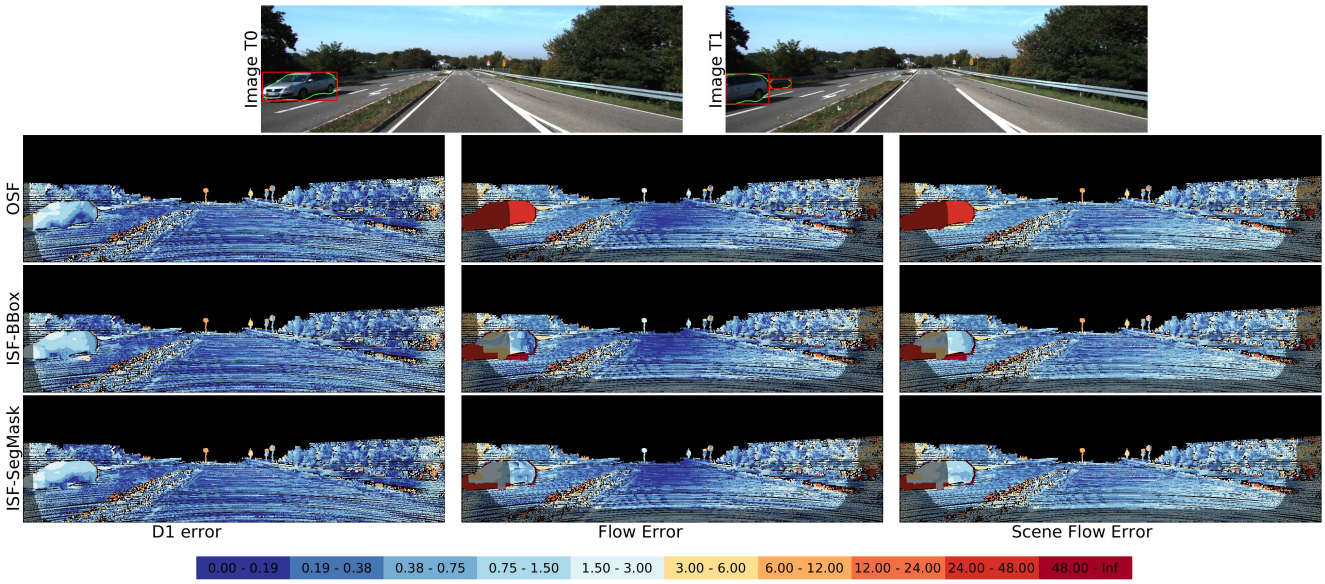

Figure 9: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

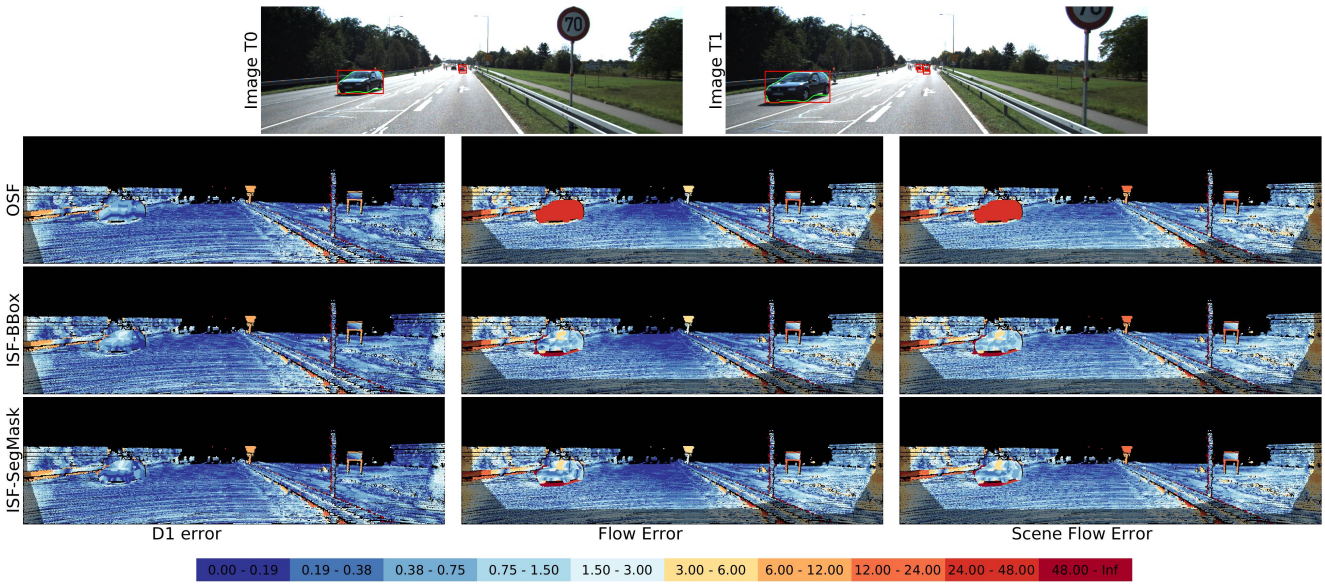

Figure 10: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

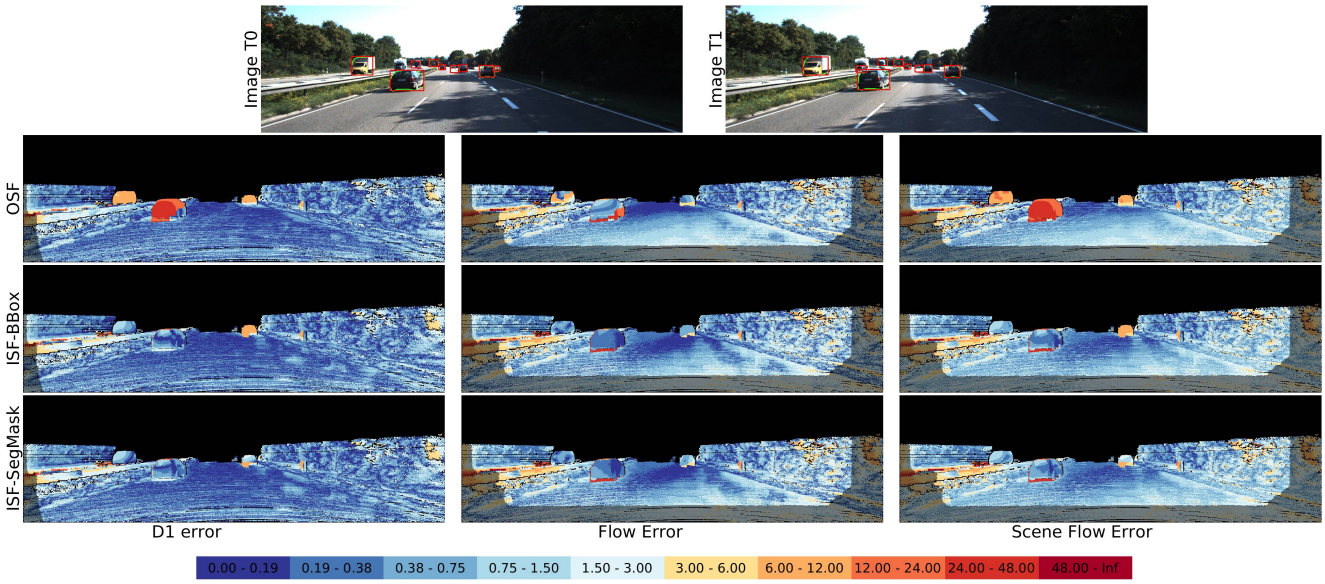

Figure 11: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

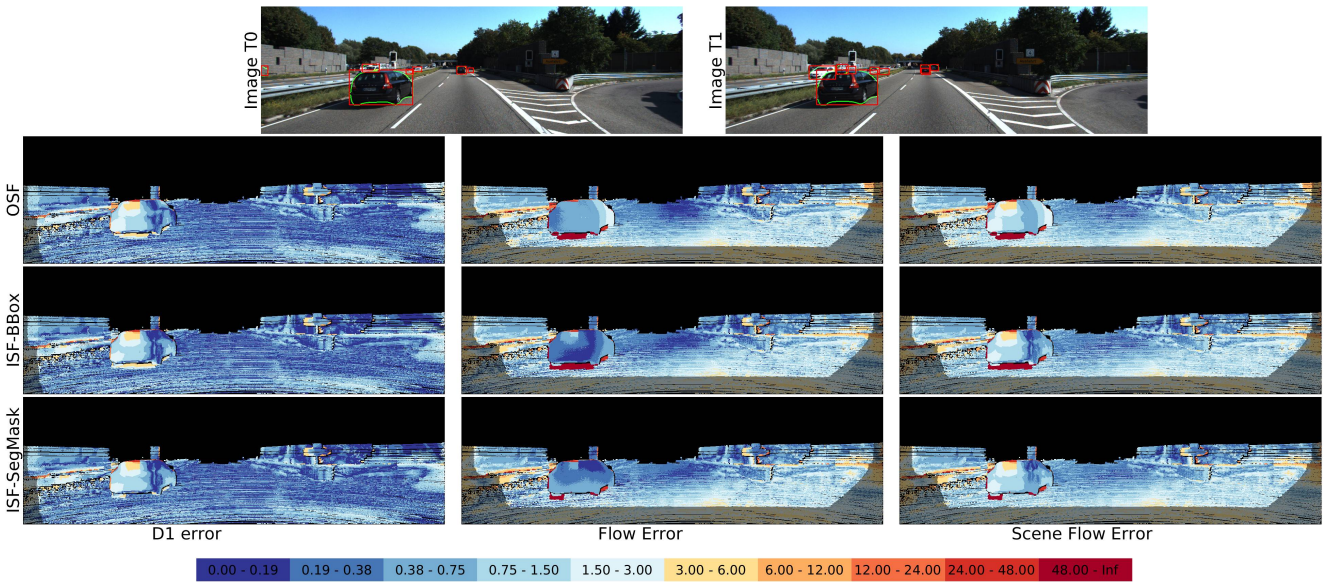

Figure 12: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

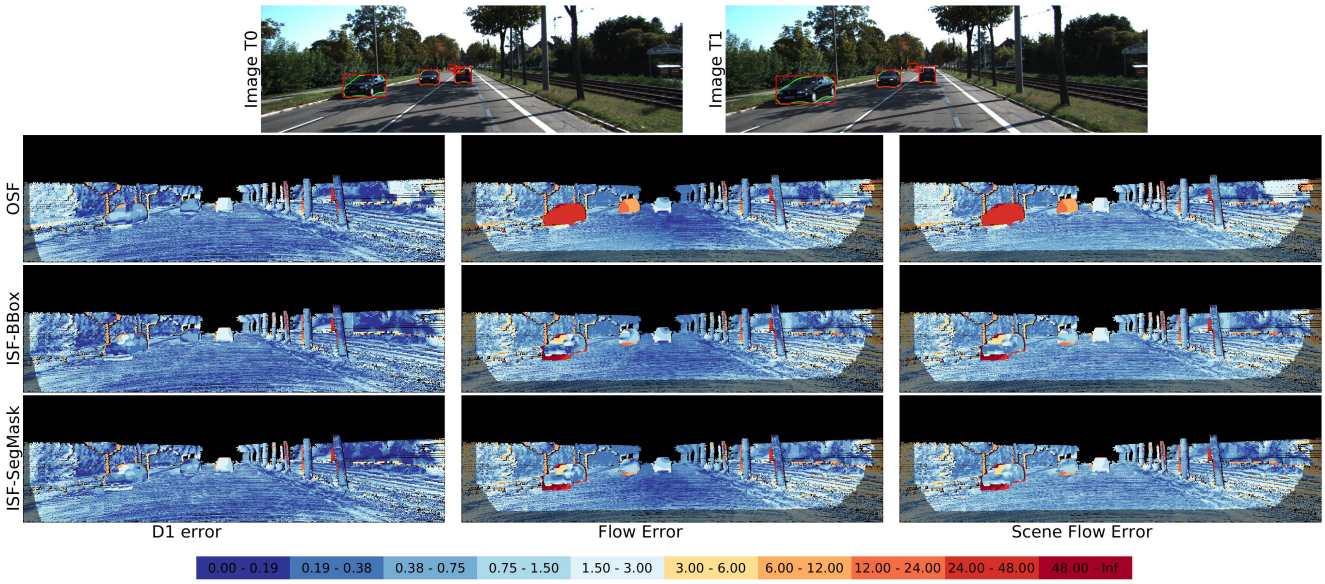

Figure 13: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

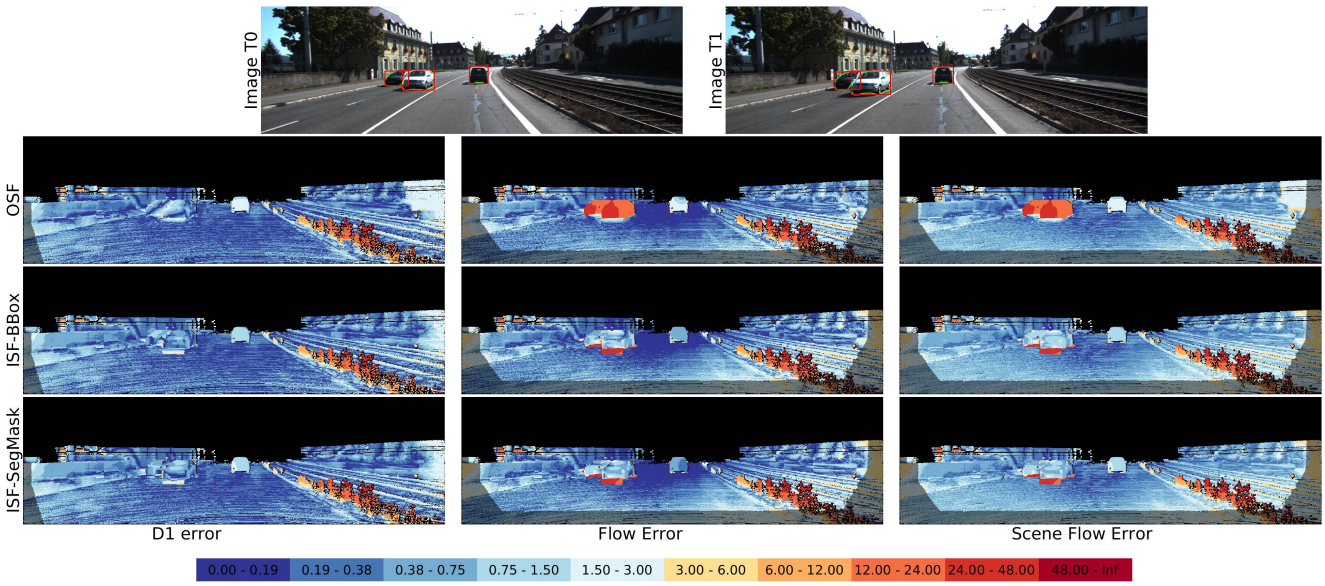

Figure 14: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

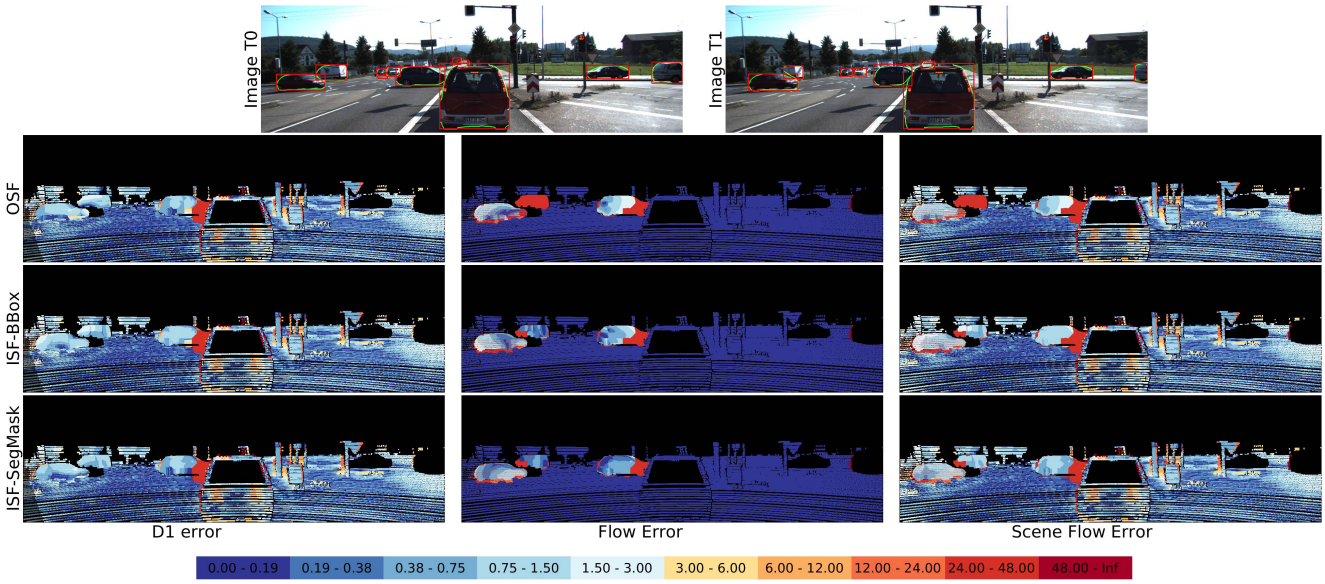

Figure 15: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

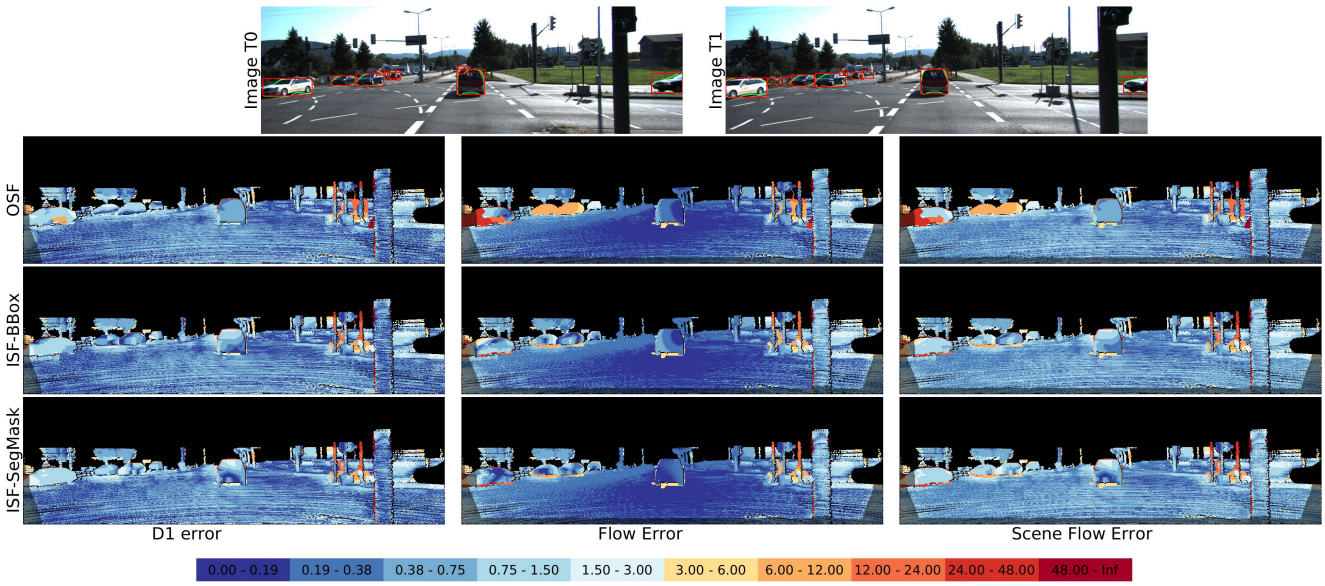

Figure 16: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

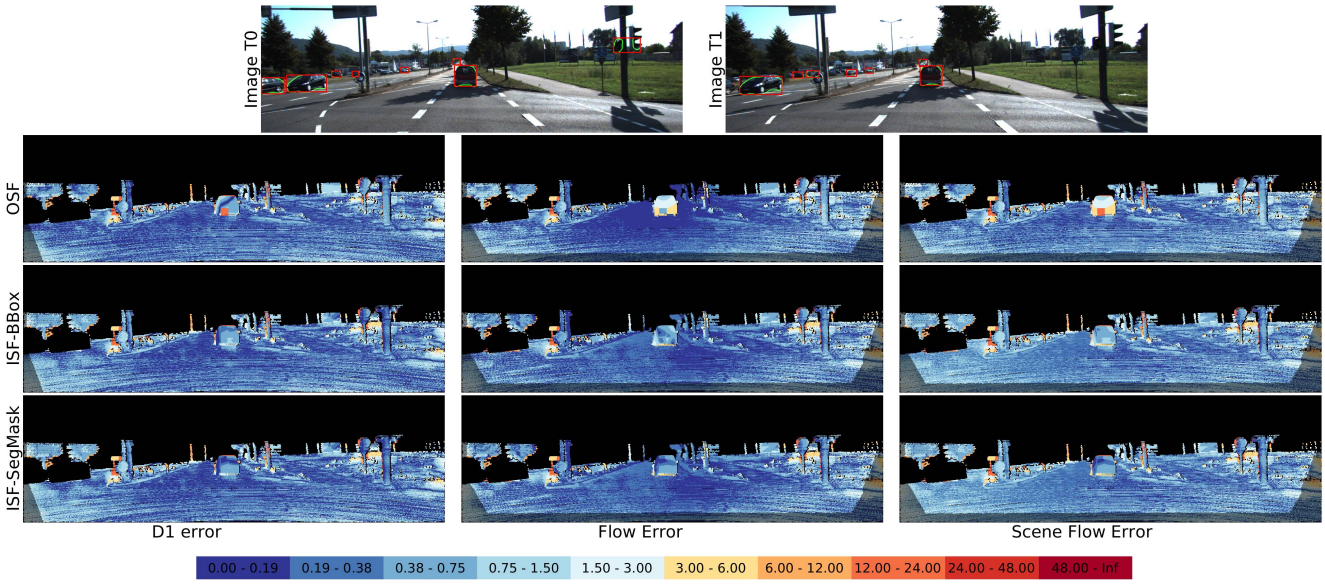

Figure 17: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

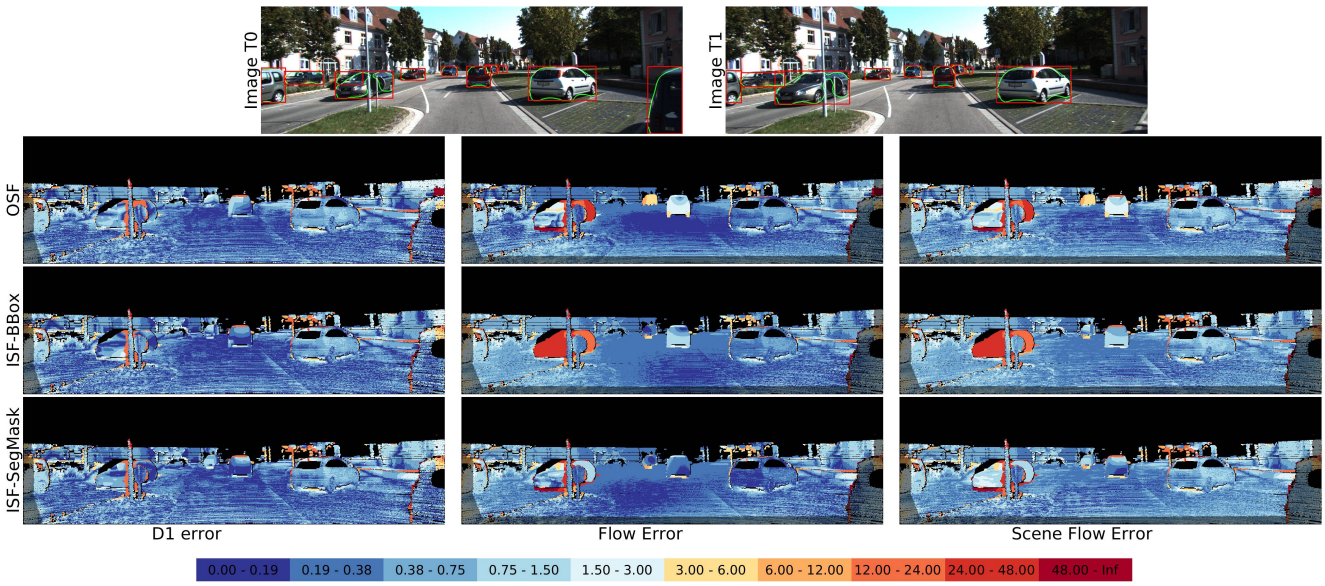

Figure 18: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

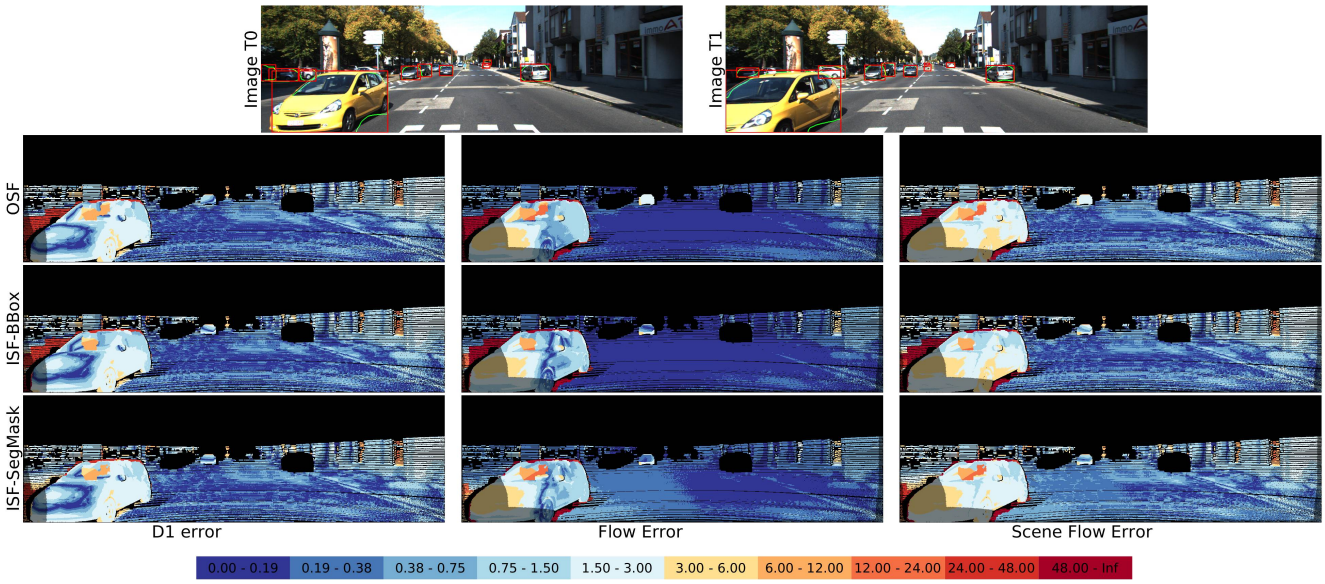

Figure 19: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

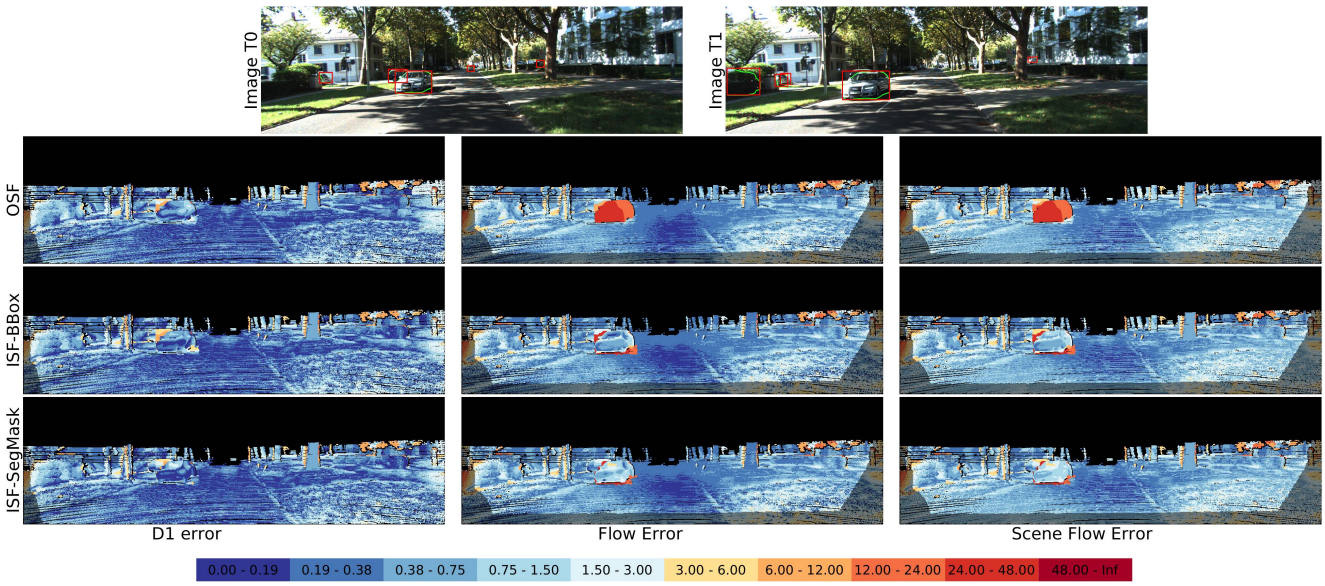

Figure 20: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

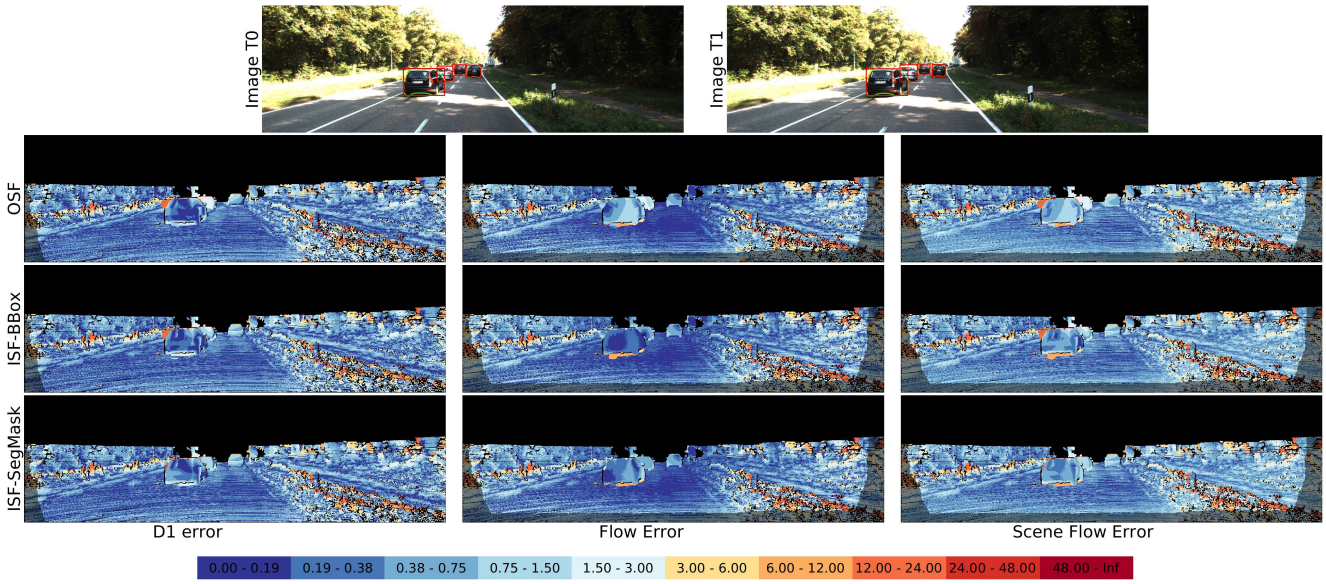

Figure 21: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

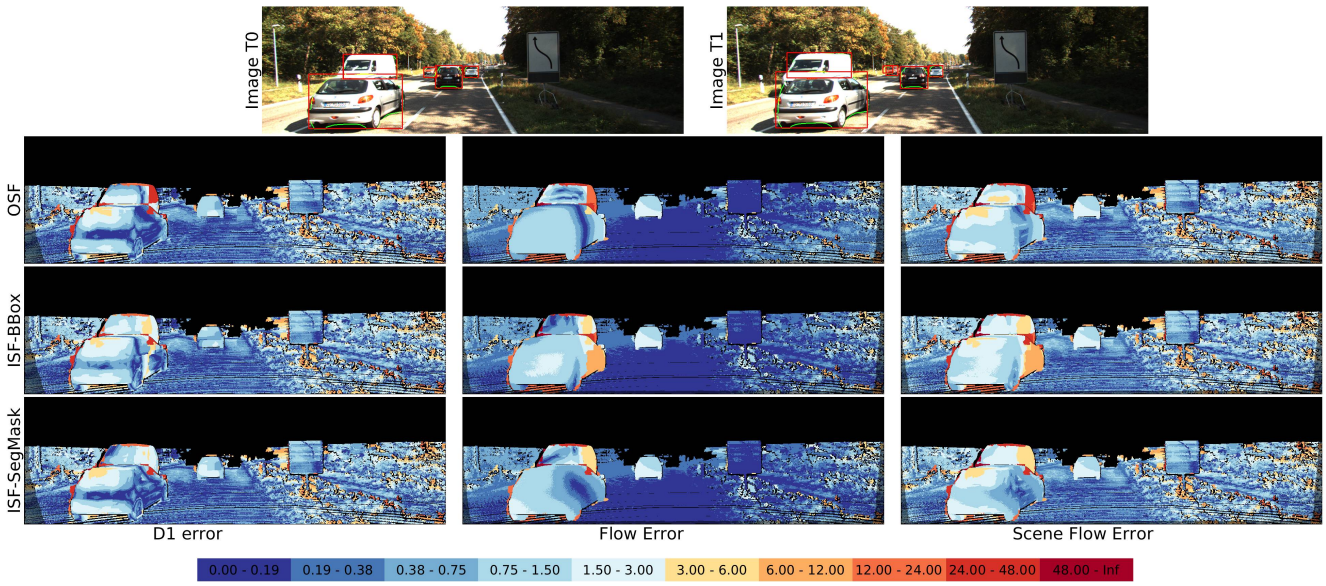

Figure 22: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

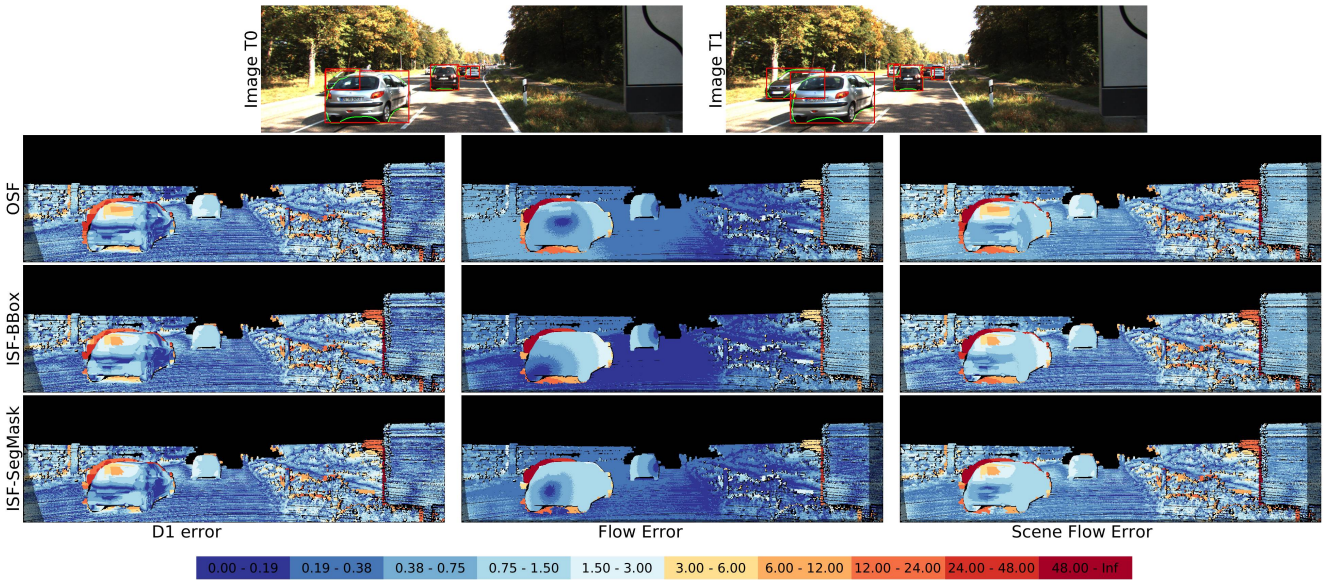

Figure 23: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

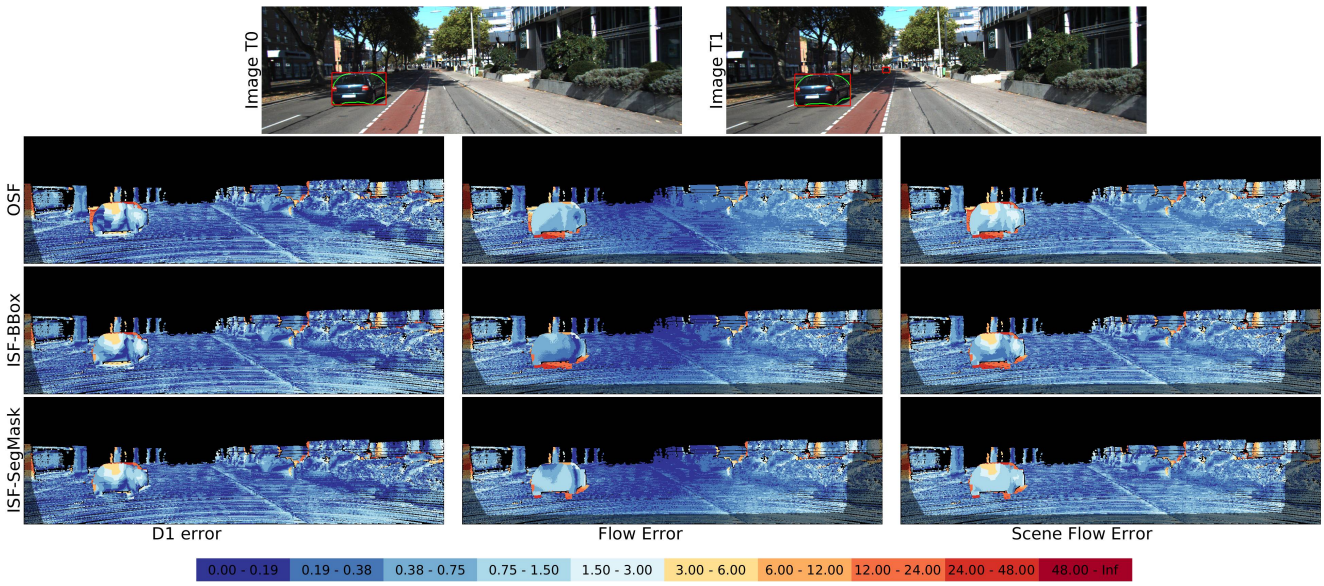

Figure 24: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

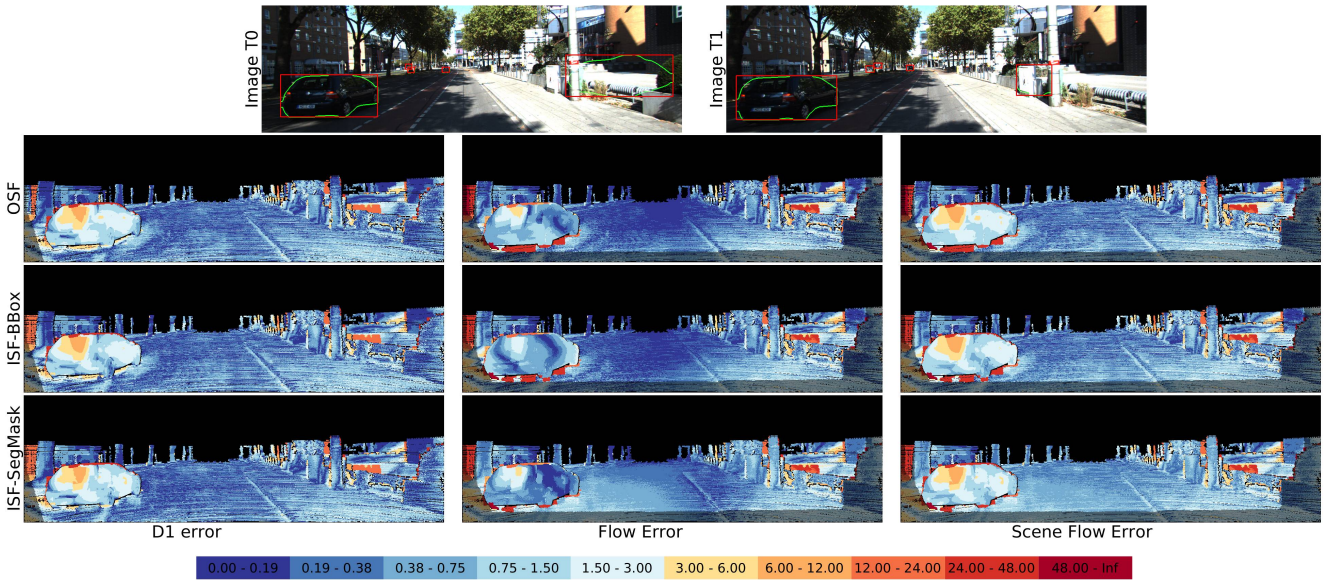

Figure 25: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

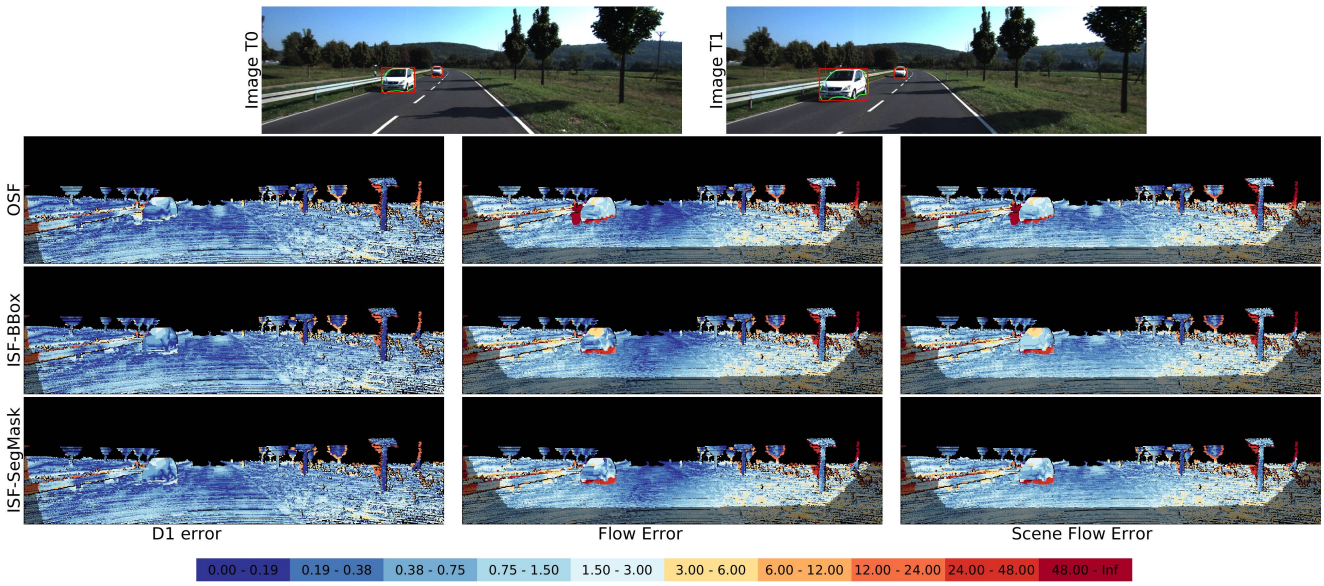

Figure 26: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

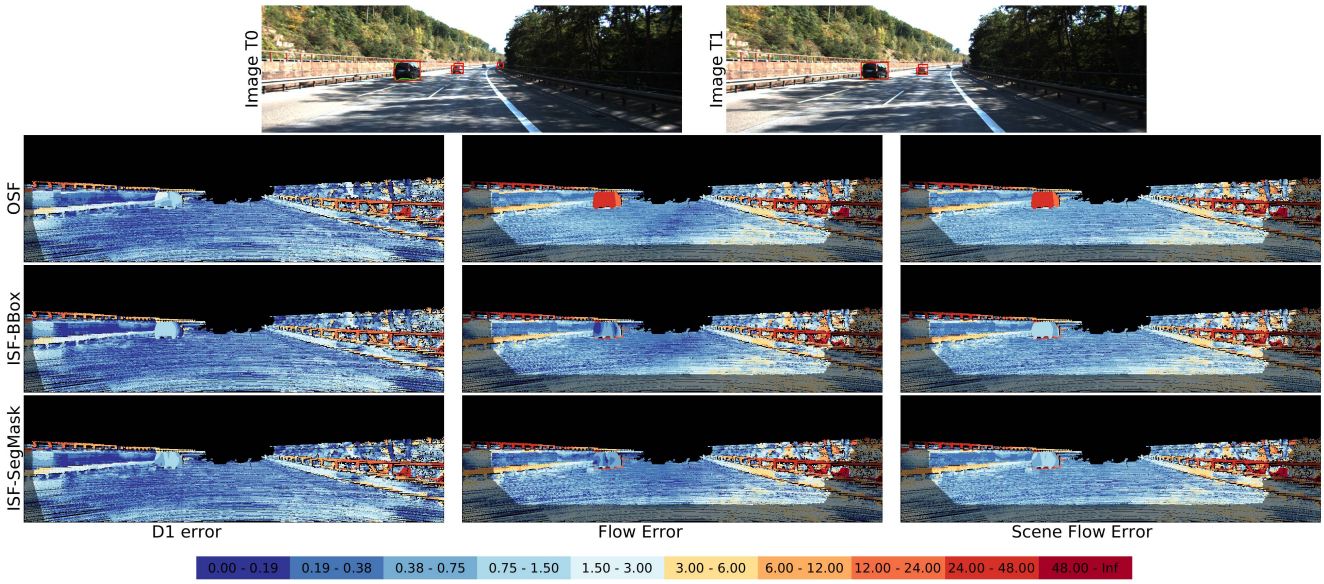

Figure 27: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

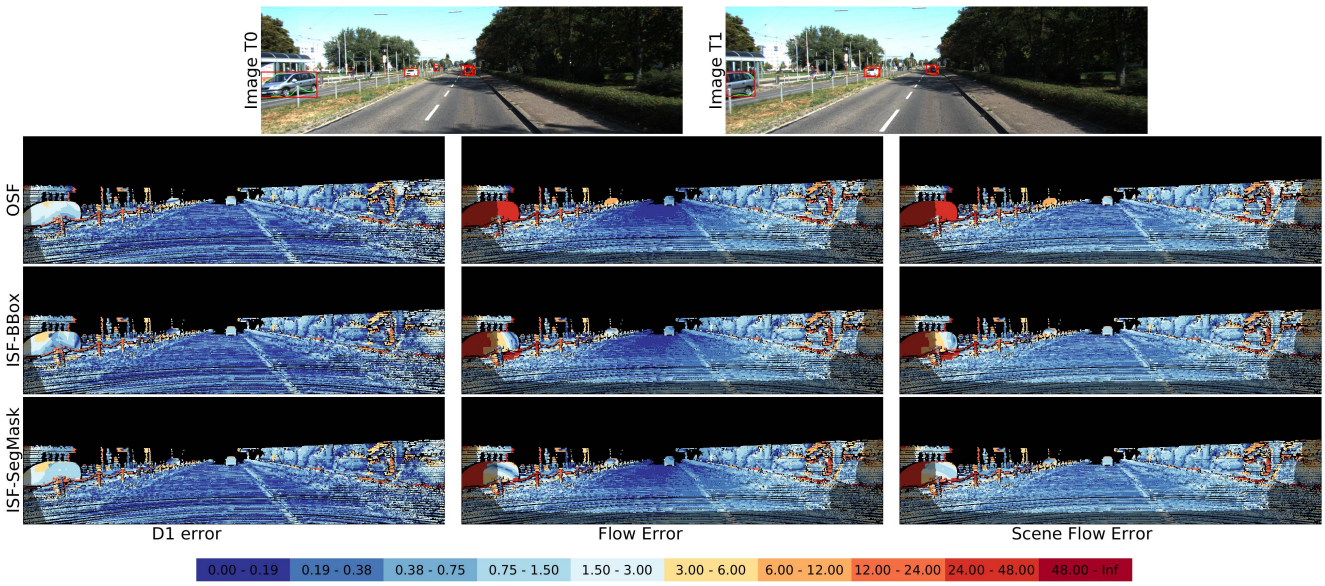

Figure 28: **Impact of Recognition Granularity on 3D Scene Flow Estimation on KITTI 2015 Validation Set.** The top row shows two consecutive input images of the left camera along with the predicted 2D bounding boxes (red) and 2D instance segmentation masks (green) which form the input to our method. The figures below show (from top-to-bottom) the results of OSF [4] (no recognition input), *ISF-BBox* (bounding box input), *ISF-SegMask* (segmentation input) in terms of disparity errors (left), optical flow errors (middle) and scene flow errors (right) using the color scheme depicted in the legend.

2. Why are 3D object coordinates not important?

In this section, we study the effect of using 3D object coordinate cues to improve 3D scene flow estimation. Towards this goal, we evaluated our *ISF-SegMask-ObjCoord* method which includes a data term in the CRF penalizing the difference in 3D object coordinates between corresponding image locations as described in the main paper. Figure 29 shows the influence of the 3D object coordinate term weight on the scene flow error. Surprisingly, we observe an increase in 3D scene flow error, in particularly for the foreground parts of the image, when increasing the importance of the 3D object coordinate cue.

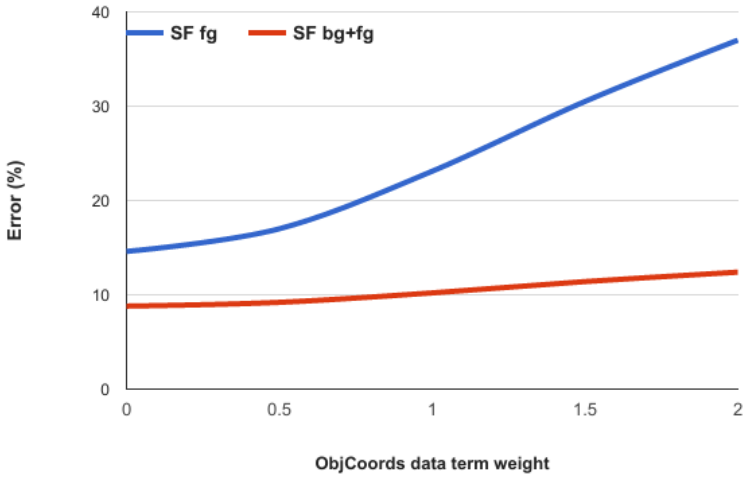

Figure 29: **Impact of 3D Object Coordinates.** This figure shows the 3D scene flow error evaluated at all pixels in the image (red) and only the foreground pixels (blue) with respect to the weight of the object coordinates data term.

We now analyze the reason why 3D object coordinates fail to improve 3D scene flow estimation beyond simpler 2D instance segmentation cues. We hypothesize that the accuracy of 3D object coordinate predictions from modern CNN-based methods is below what is required to further improve 3D scene flow estimation. In order to validate this hypothesis, we computed optical flow for the foreground part of the scene by finding correspondences between the reference and target frame using the 3D object coordinates. We then compare this flow with the optical flow results of our CRF-based *ISF-SegMask* method after initialization and with the final results after optimization. Quantitatively, we find that the optical flow computed via 3D object coordinate matching improves upon *ISF-SegMask* initialization and final flow output for only 1.4% and 1% of the foreground pixels, respectively. For the remaining pixels, however, the quality of the 3D object coordinates does not reach the level required for accurate 3D scene flow estimation. Fig. 30-47 illustrate this observation on a few examples from our validation set. The figures show that flow accuracy from object coordinate matching is either worse or similar at places where *ISF-SegMask* fails. Therefore, 3D object coordinates fail in improving 3D scene flow estimation using our CRF framework.

Remark: We compare the quality of our 3D object coordinate predictions to state-of-the-art approaches using random forests in the last section of this supplementary and demonstrate significantly better performance when using our convolutional neural network architecture. This demonstrates that even state-of-the-art object coordinate predictions are not helpful for improving 3D scene flow due to the high level of accuracy required by current benchmarks (i.e., 3 pixels error in KITTI).

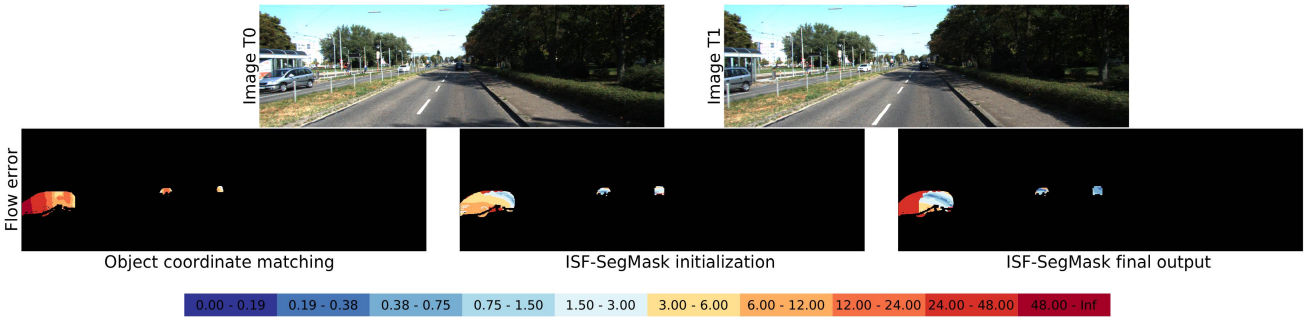

Figure 30: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

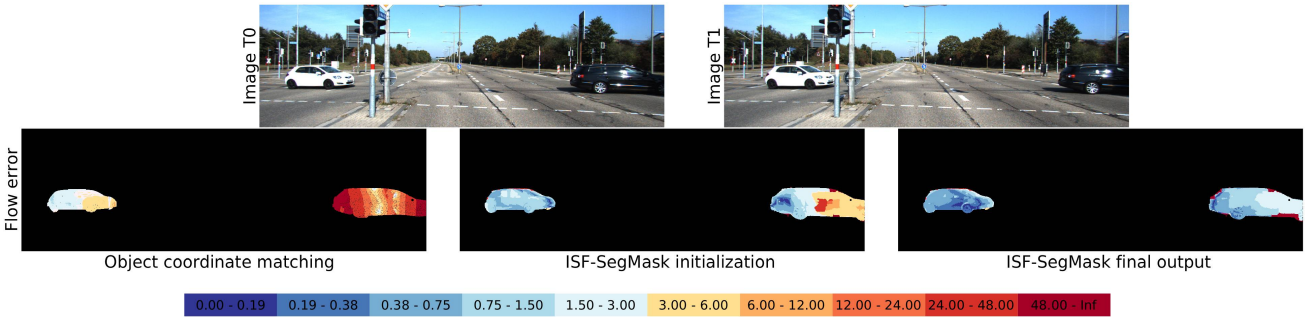

Figure 31: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

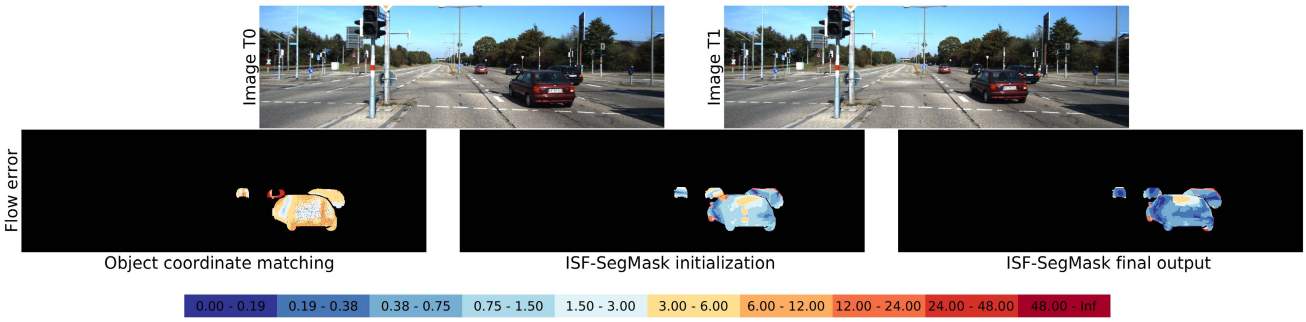

Figure 32: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

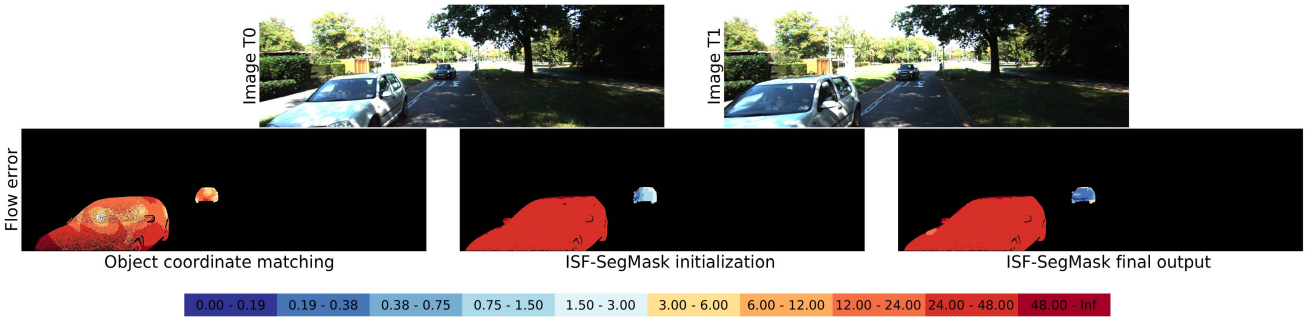

Figure 33: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

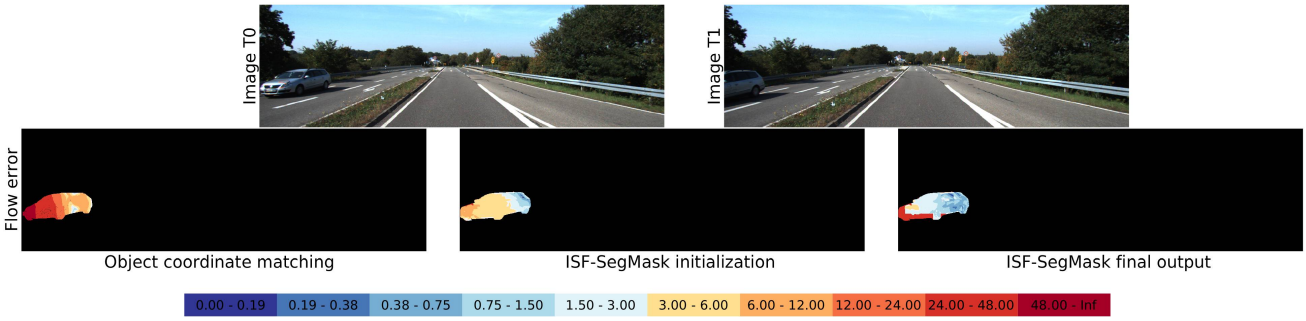

Figure 34: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

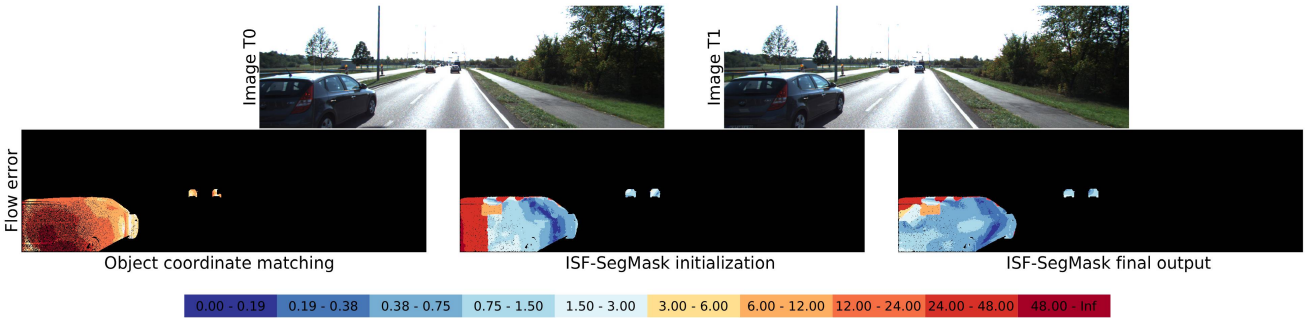

Figure 35: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

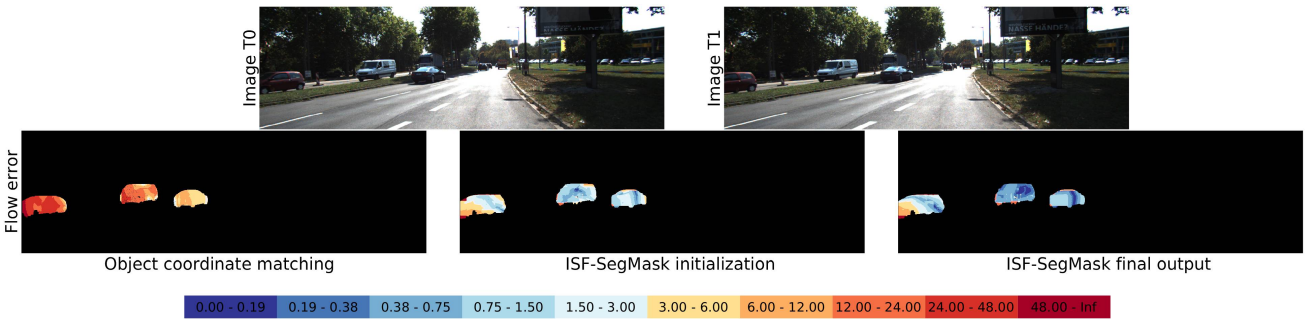

Figure 36: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

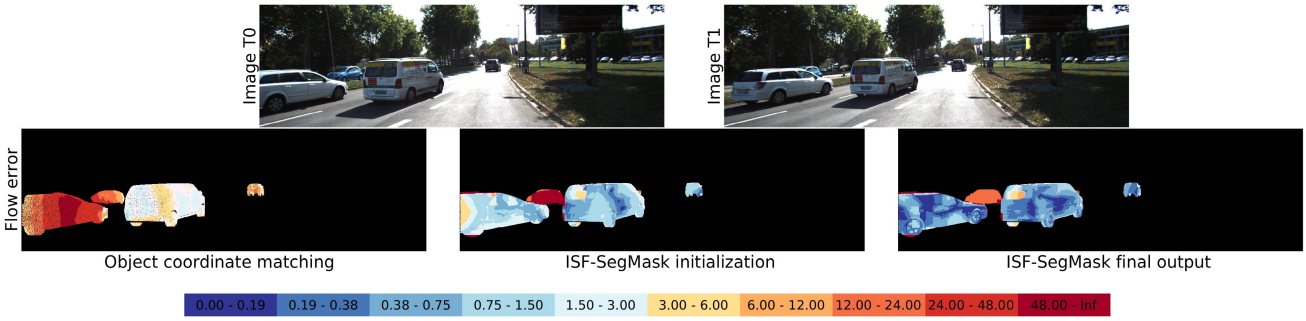

Figure 37: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

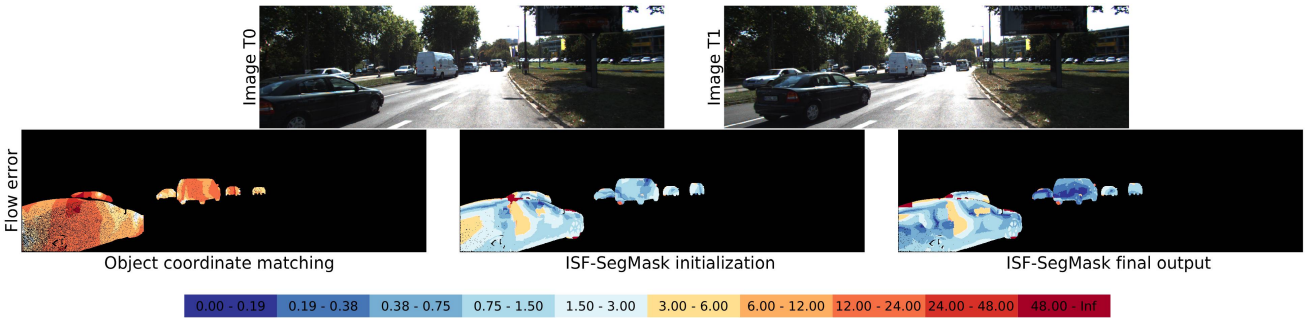

Figure 38: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

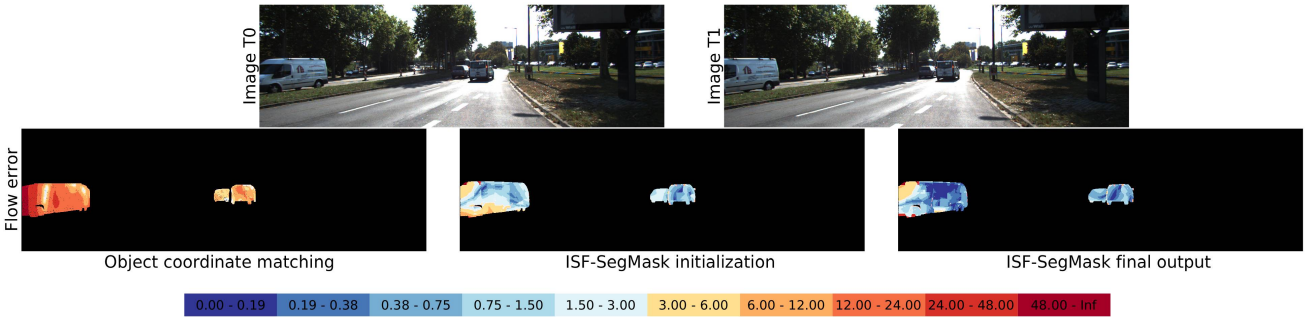

Figure 39: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

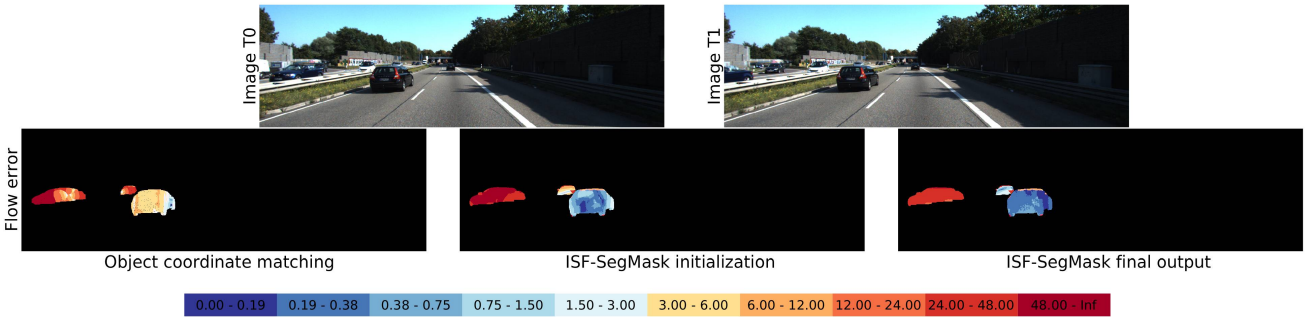

Figure 40: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

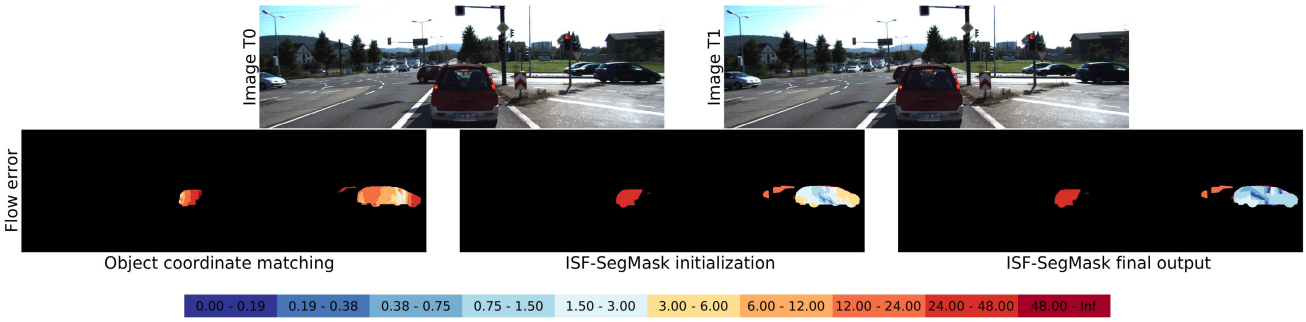

Figure 41: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

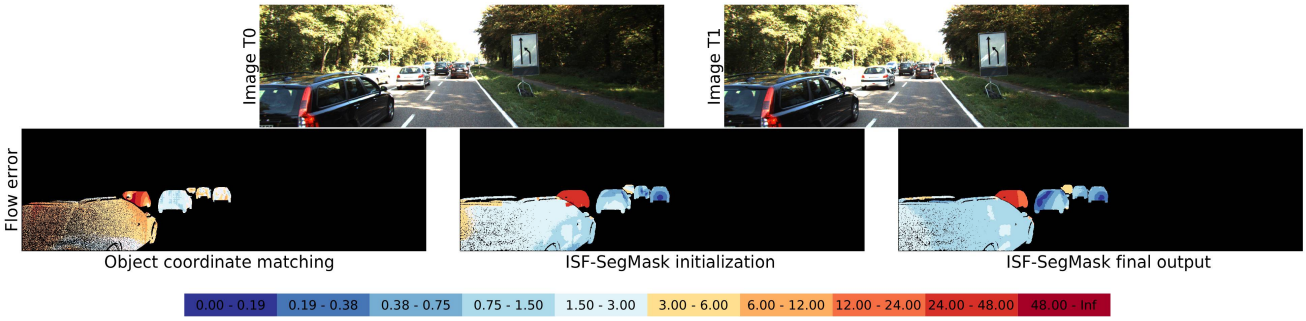

Figure 42: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

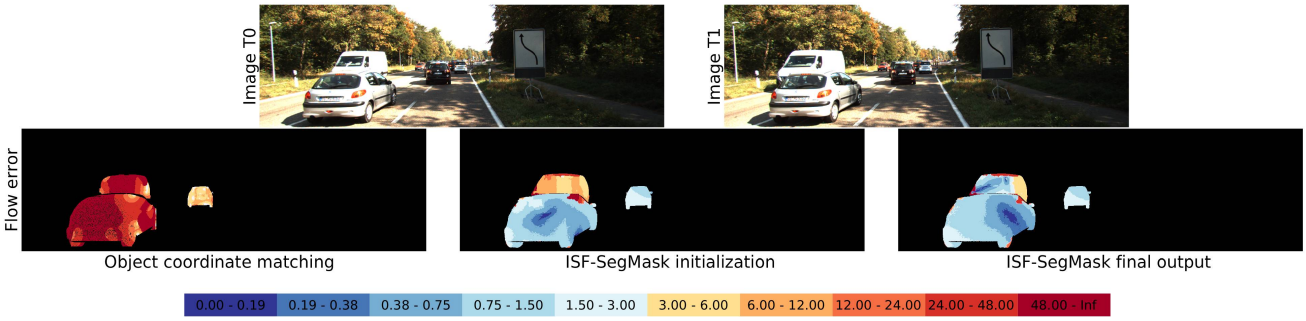

Figure 43: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

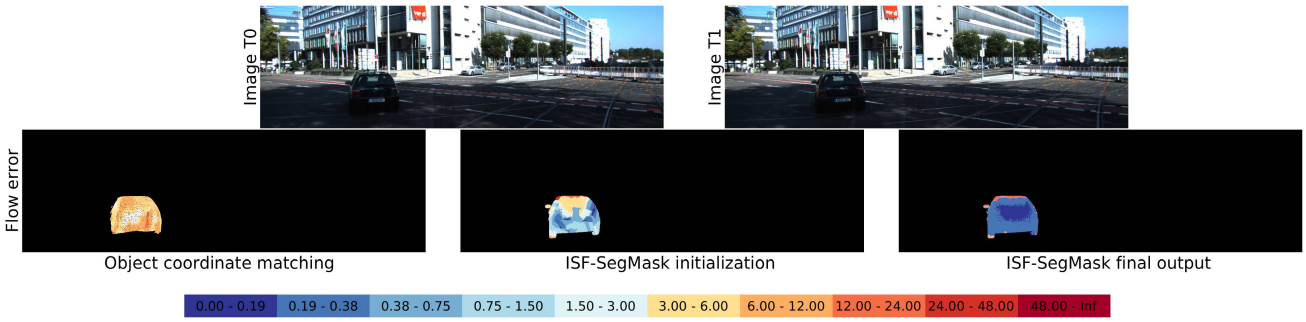

Figure 44: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

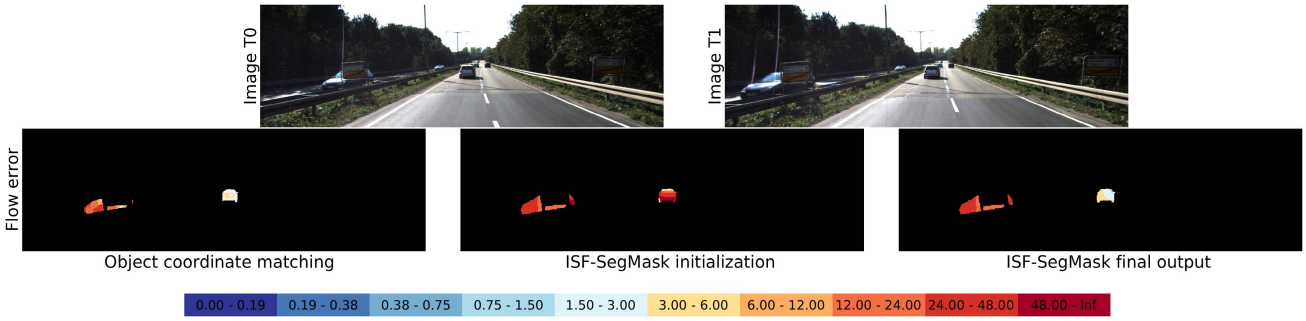

Figure 45: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

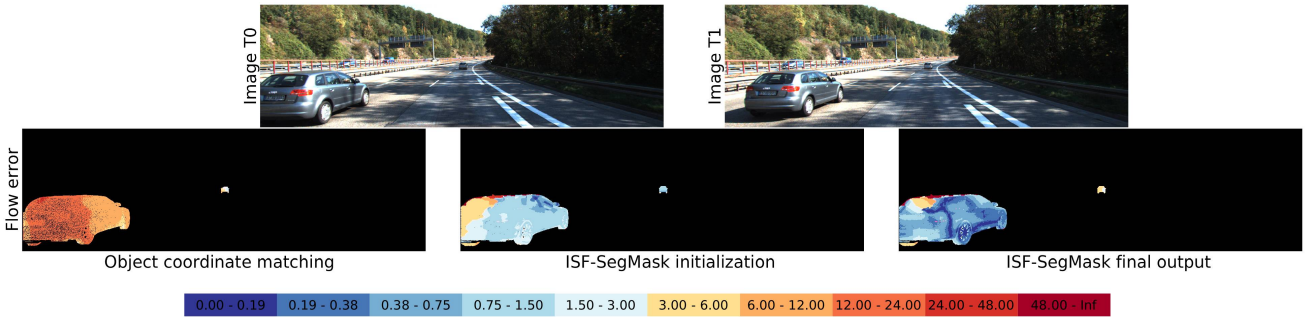

Figure 46: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

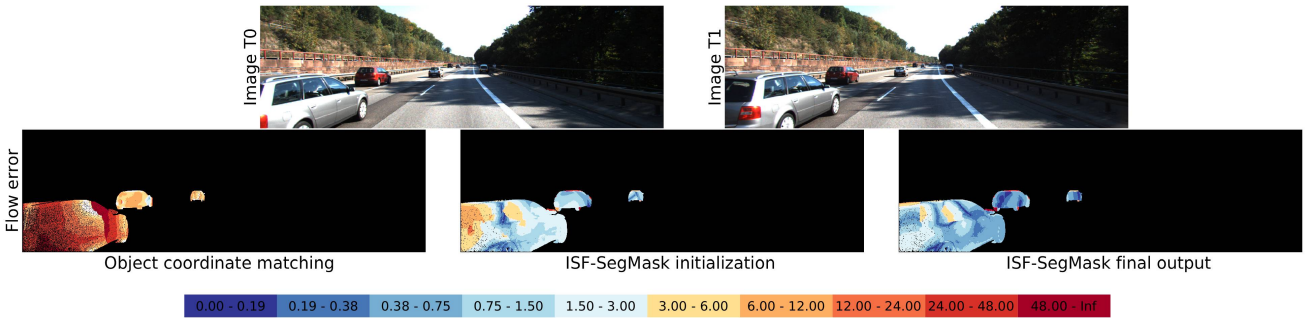

Figure 47: **Qualitative Analysis of Flow from 3D Object Coordinate on KITTI 2015 Validation Set.** The top row shows two consecutive frames of the left camera. Each figure in the bottom row shows optical flow errors on foreground pixels in the reference view using the color scheme depicted in the legend. In particular, we compare results when flow is obtained directly from the 3D object coordinate predictions (left) to our *ISF-SegMask* method before (middle) and after (right) optimization.

3. Qualitative Comparison on KITTI 2015 Test Set

In this section, we provide additional qualitative comparisons of our method (*ISF-SegMask*) to other state-of-the-art methods (OSF [4], PRSM [7], OSF-TC [5]) which we obtained from the KITTI 2015 scene flow evaluation server. For each scene and method, we show the input image sequence (top), the error images corresponding to the estimated disparity (left), optical flow (middle) and scene flow (right) using a logarithmic color coding where red shades represent errors above 3 pixels / 5% relative error and blue shades denote errors below 3 pixels / 5% relative error.

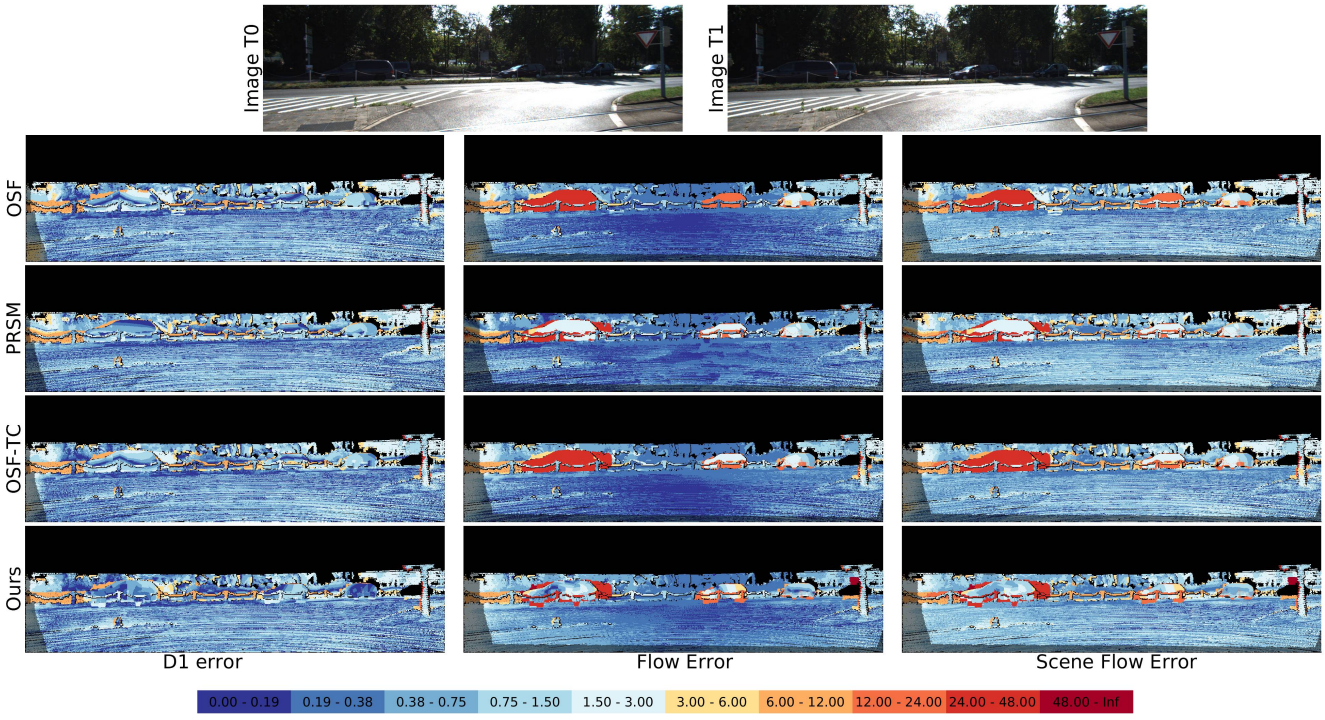

Figure 48: **Qualitative Comparison of Our Method to the State-of-the-Art on the KITTI 2015 Test Set.** The top row shows the input images. Each figure shows (from top-to-bottom) the results of OSF [4], PRSM [7], OSF-TC [5], and *ISF-SegMask* respectively, using a logarithmic color coding where red shades represent errors above 3 pixels / 5% relative error and blue shades denote errors below 3 pixels / 5% relative error.

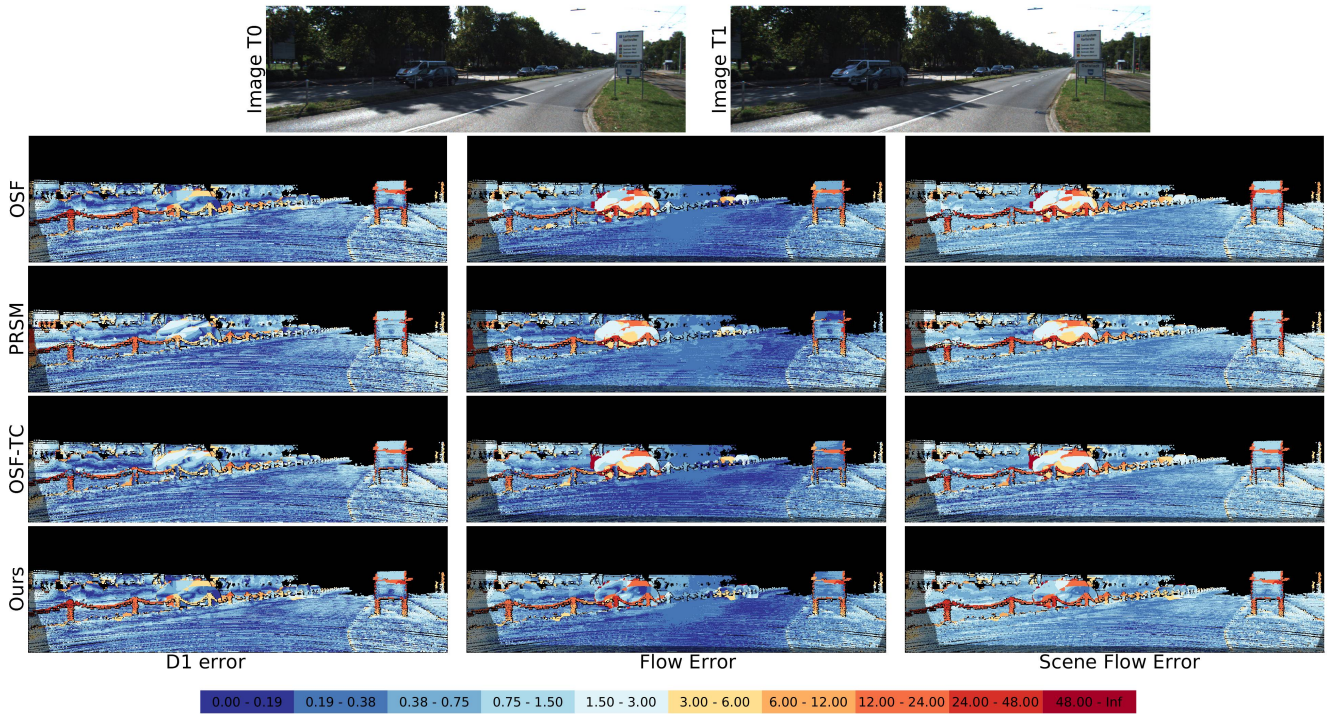

Figure 49: **Qualitative Comparison of Our Method to the State-of-the-Art on the KITTI 2015 Test Set.** The top row shows the input images. Each figure shows (from top-to-bottom) the results of OSF [4], PRSM [7], OSF-TC [5], and *ISF-SegMask* respectively, using a logarithmic color coding where red shades represent errors above 3 pixels / 5% relative error and blue shades denote errors below 3 pixels / 5% relative error.

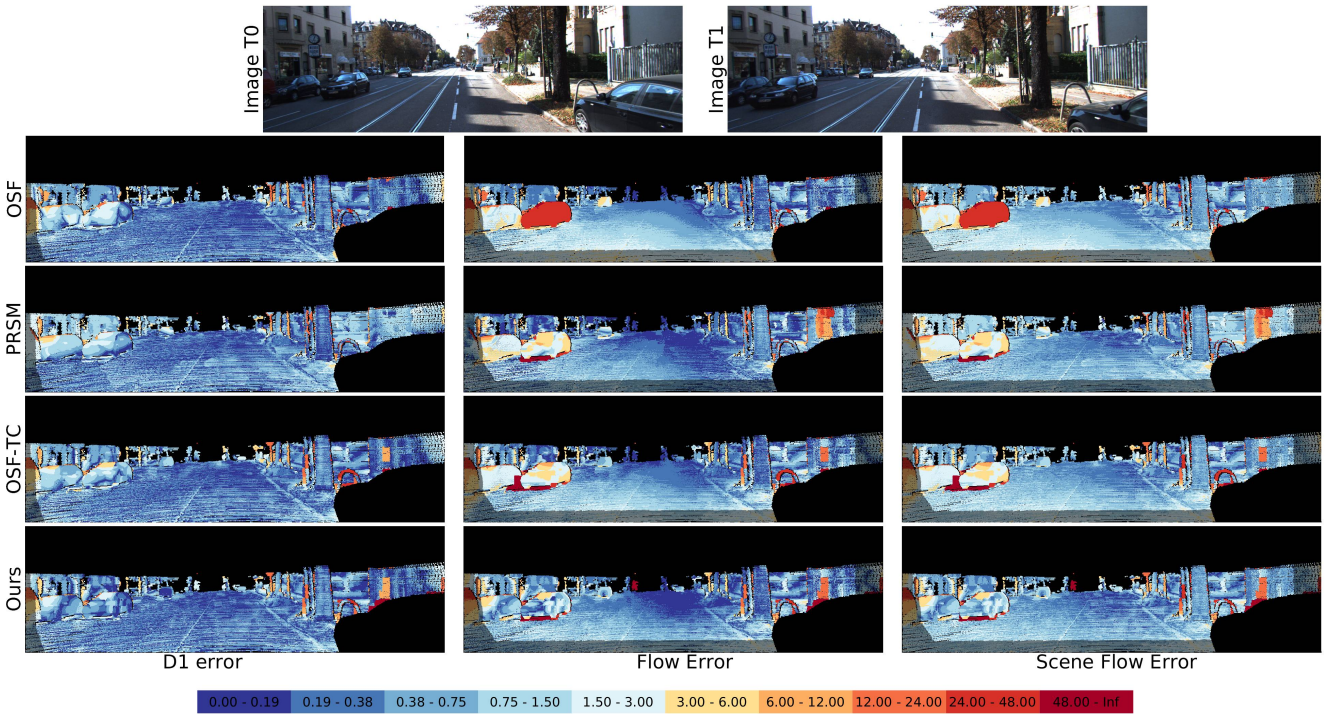

Figure 50: **Qualitative Comparison of Our Method to the State-of-the-Art on the KITTI 2015 Test Set.** The top row shows the input images. Each figure shows (from top-to-bottom) the results of OSF [4], PRSM [7], OSF-TC [5], and *ISF-SegMask* respectively, using a logarithmic color coding where red shades represent errors above 3 pixels / 5% relative error and blue shades denote errors below 3 pixels / 5% relative error.

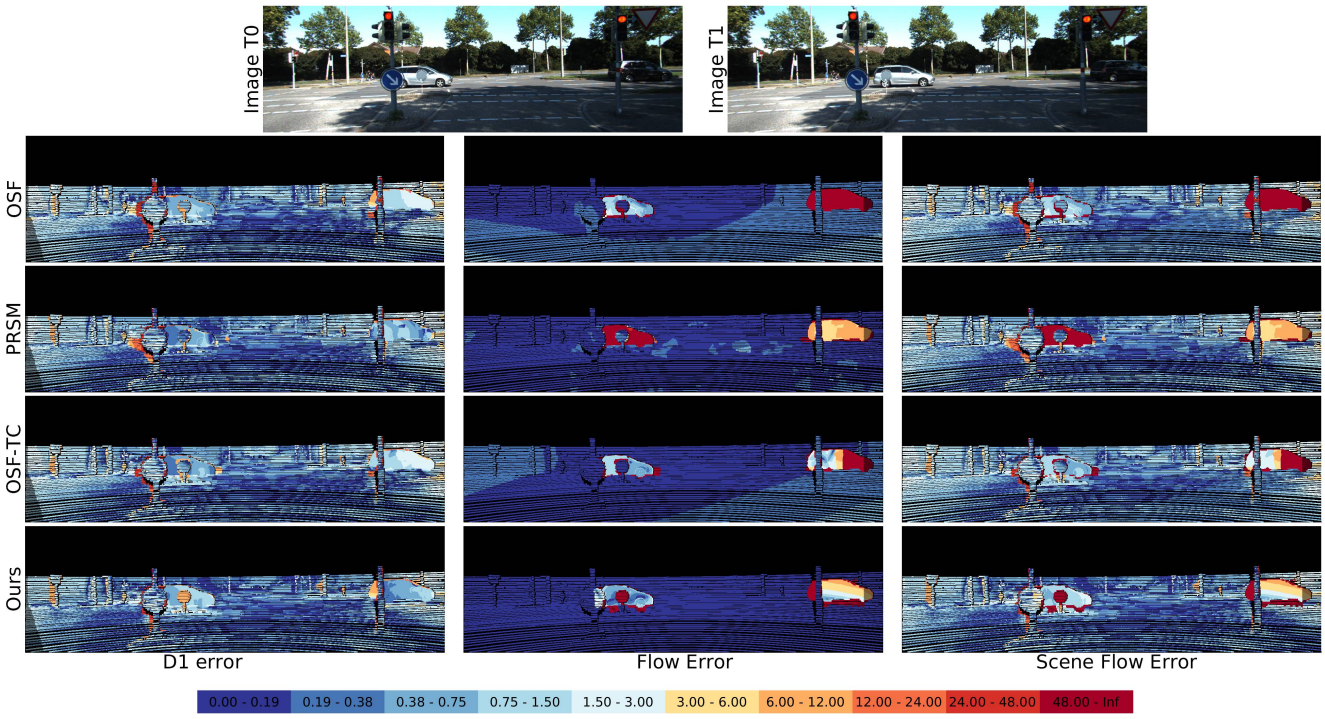

Figure 51: **Qualitative Comparison of Our Method to the State-of-the-Art on the KITTI 2015 Test Set.** The top row shows the input images. Each figure shows (from top-to-bottom) the results of OSF [4], PRSM [7], OSF-TC [5], and *ISF-SegMask* respectively, using a logarithmic color coding where red shades represent errors above 3 pixels / 5% relative error and blue shades denote errors below 3 pixels / 5% relative error.

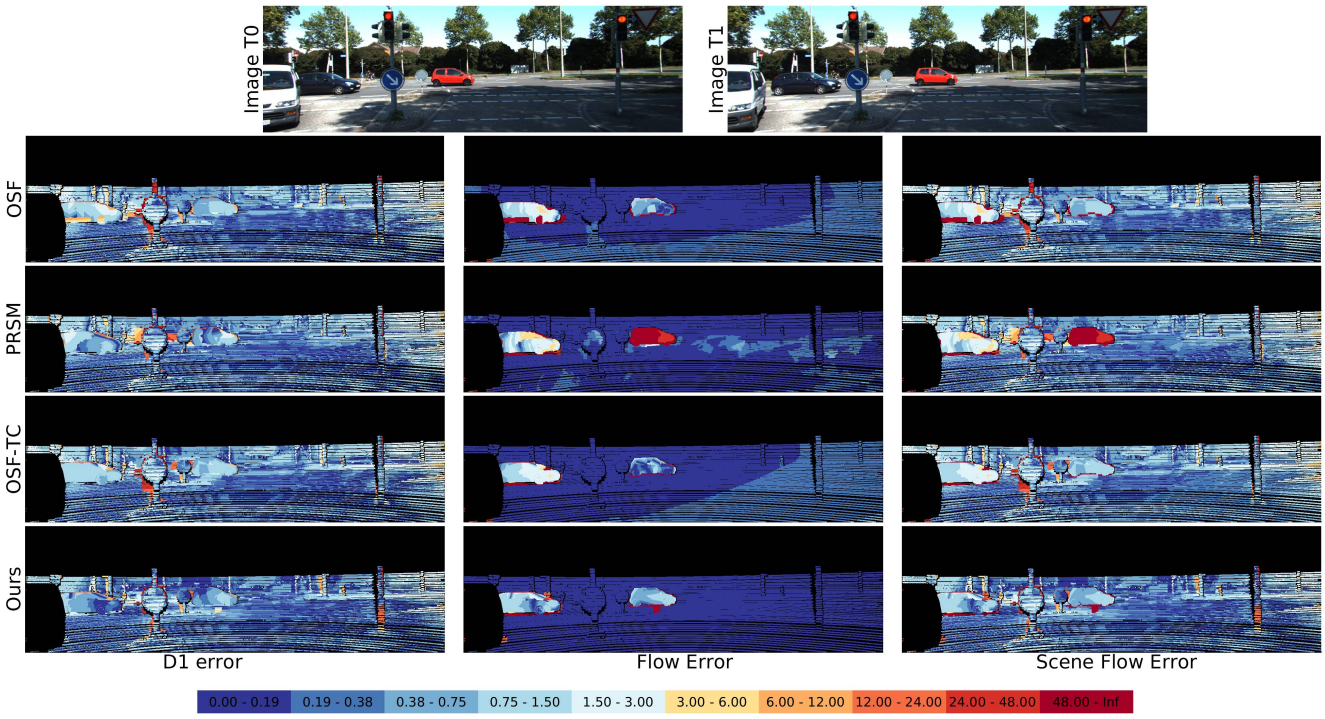

Figure 52: **Qualitative Comparison of Our Method to the State-of-the-Art on the KITTI 2015 Test Set.** The top row shows the input images. Each figure shows (from top-to-bottom) the results of OSF [4], PRSM [7], OSF-TC [5], and *ISF-SegMask* respectively, using a logarithmic color coding where red shades represent errors above 3 pixels / 5% relative error and blue shades denote errors below 3 pixels / 5% relative error.

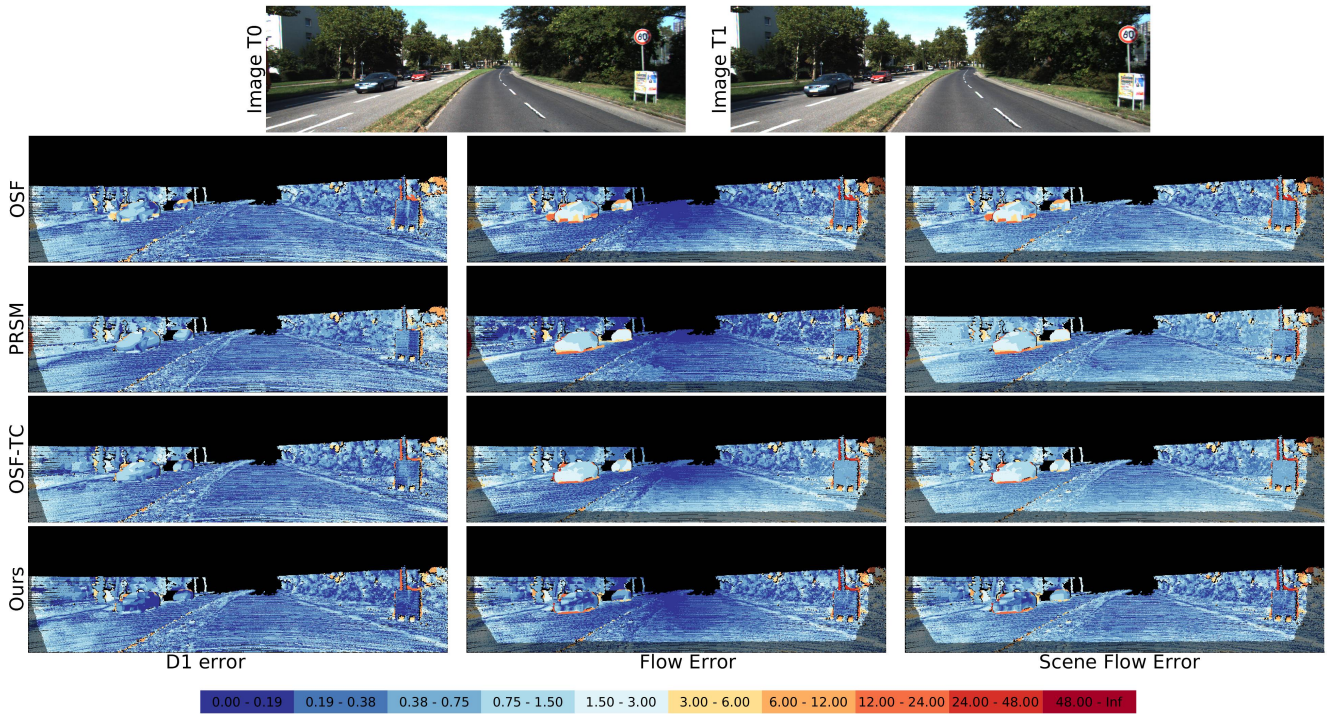

Figure 53: **Qualitative Comparison of Our Method to the State-of-the-Art on the KITTI 2015 Test Set.** The top row shows the input images. Each figure shows (from top-to-bottom) the results of OSF [4], PRSM [7], OSF-TC [5], and *ISF-SegMask* respectively, using a logarithmic color coding where red shades represent errors above 3 pixels / 5% relative error and blue shades denote errors below 3 pixels / 5% relative error.

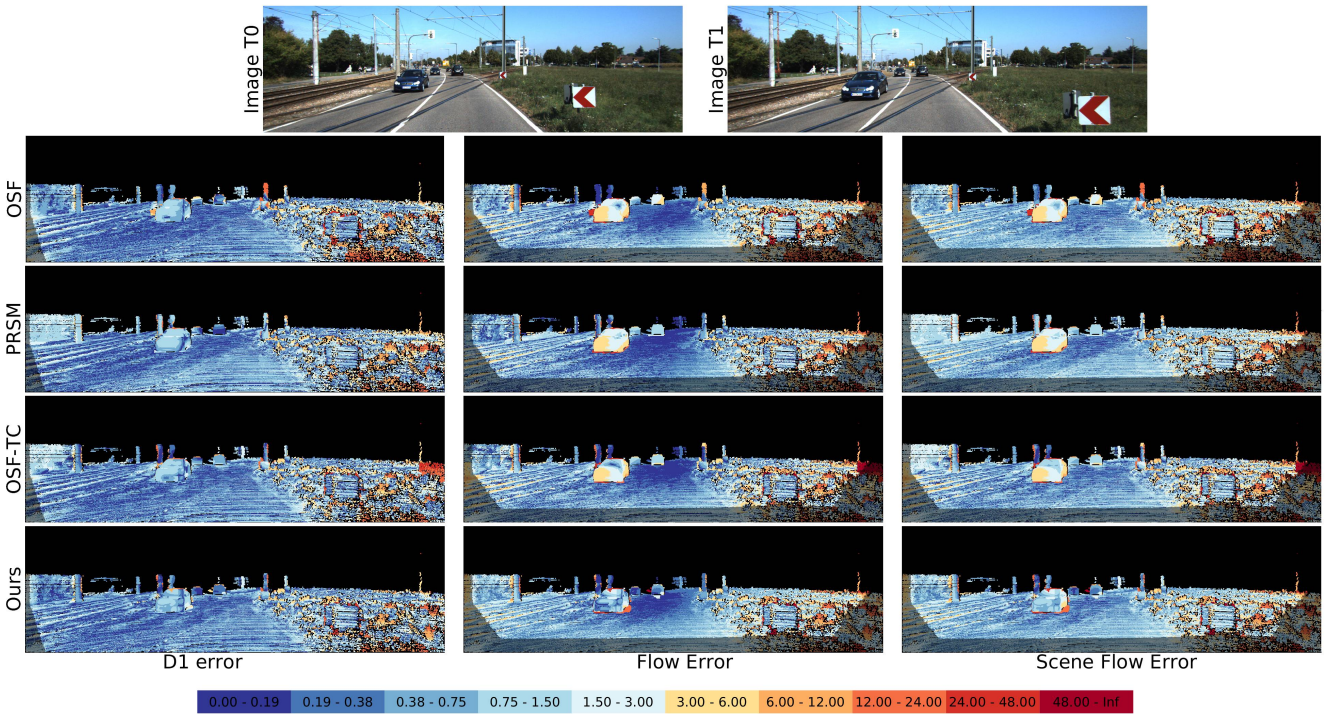

Figure 54: **Qualitative Comparison of Our Method to the State-of-the-Art on the KITTI 2015 Test Set.** The top row shows the input images. Each figure shows (from top-to-bottom) the results of OSF [4], PRSM [7], OSF-TC [5], and *ISF-SegMask* respectively, using a logarithmic color coding where red shades represent errors above 3 pixels / 5% relative error and blue shades denote errors below 3 pixels / 5% relative error.

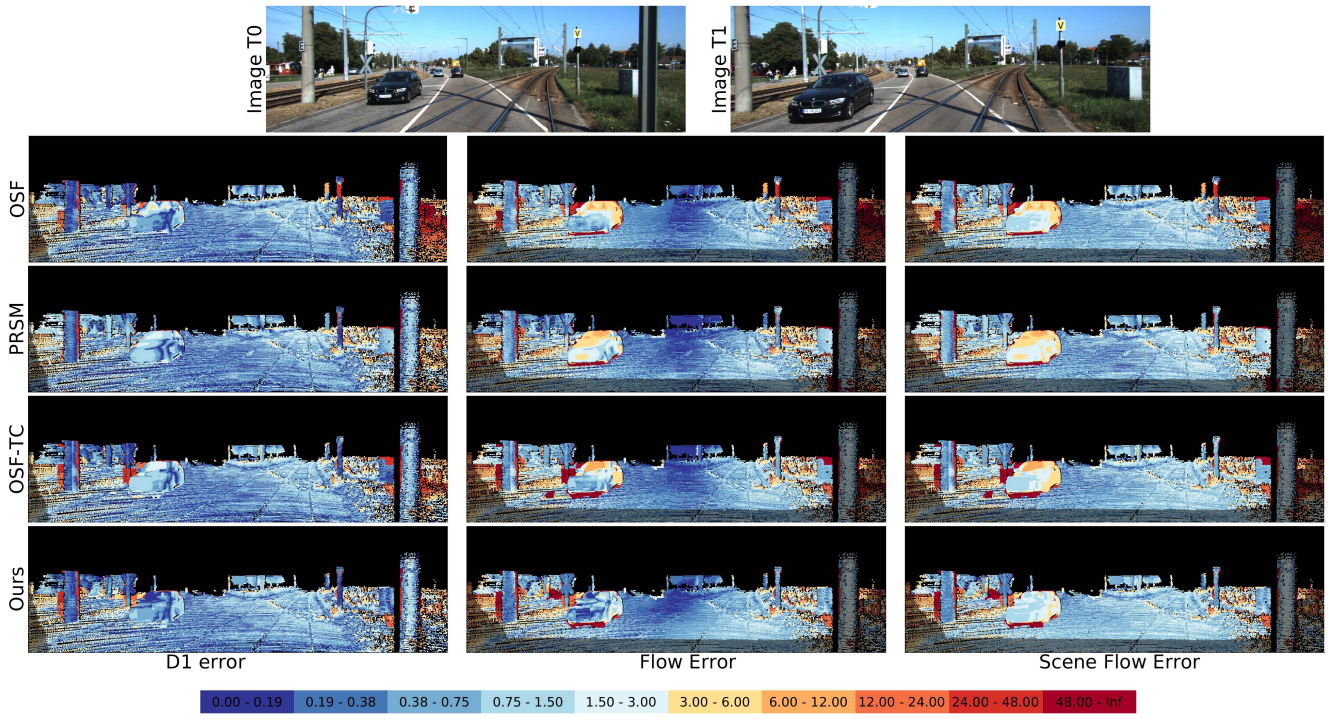

Figure 55: **Qualitative Comparison of Our Method to the State-of-the-Art on the KITTI 2015 Test Set.** The top row shows the input images. Each figure shows (from top-to-bottom) the results of OSF [4], PRSM [7], OSF-TC [5], and *ISF-SegMask* respectively, using a logarithmic color coding where red shades represent errors above 3 pixels / 5% relative error and blue shades denote errors below 3 pixels / 5% relative error.

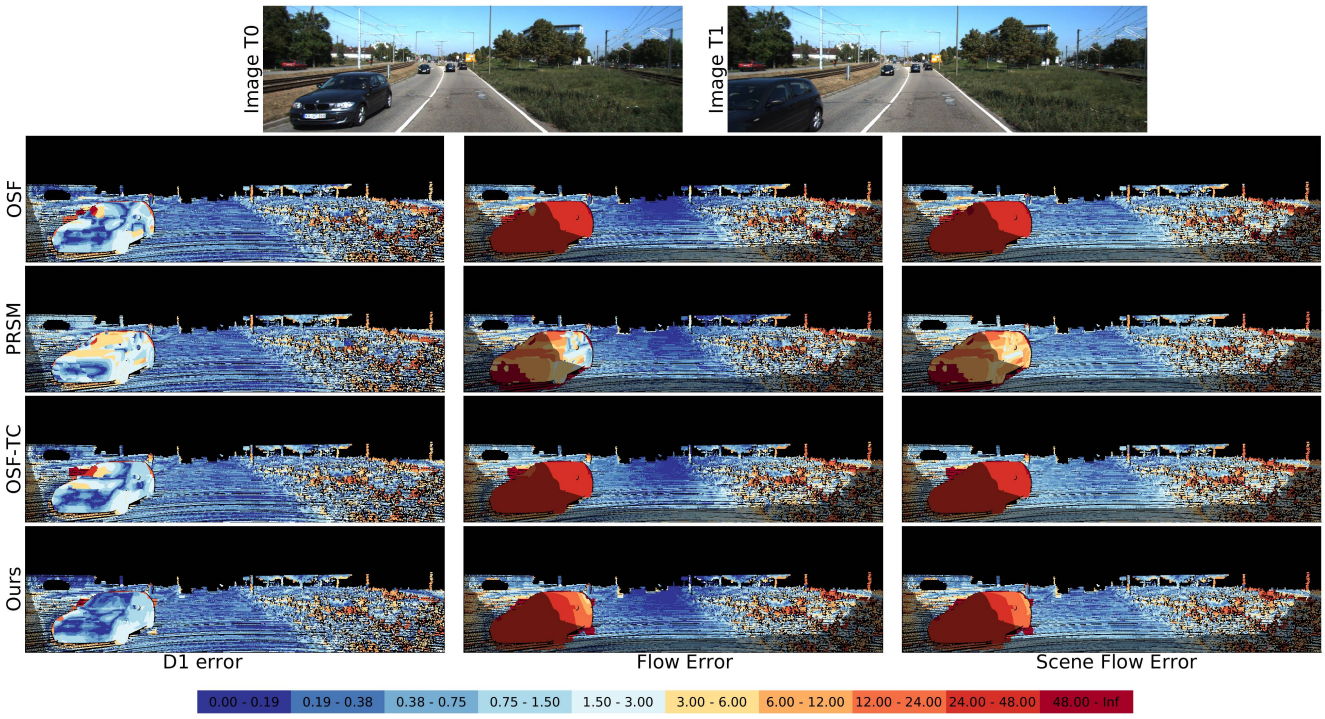

Figure 56: **Qualitative Comparison of Our Method to the State-of-the-Art on the KITTI 2015 Test Set.** The top row shows the input images. Each figure shows (from top-to-bottom) the results of OSF [4], PRSM [7], OSF-TC [5], and *ISF-SegMask* respectively, using a logarithmic color coding where red shades represent errors above 3 pixels / 5% relative error and blue shades denote errors below 3 pixels / 5% relative error.

4. 3D Object Coordinate Prediction

In this section, we illustrate and describe the CNN architecture that we have used for predicting 3D object coordinates. An illustration of our architecture can be found in Fig. 57. We use an Encoder-Decoder style network similar to that of [6]. The input to our network is the RGB image and the XYZ point cloud of a car instance that is predicted from the instance prediction network. The XYZ point cloud can be obtained using the estimated disparity and the camera calibration which is known. The network regresses the X, Y and Z values in object coordinate system for each pixel of the input. We consider this as fine grained unique geometric labeling of the object's surface points.

The encoder part of the network is a set of 5 convolutional layers with a stride of 2 at each of them (Conv0-Conv4), followed by 3 fully connected layers (FC1-FC3). Further, the FC3 layer is reshaped to a matrix again and follows through a set of deconvolutional layers (Dec4-Dec0). Each convolutional/deconvolutional layer is follwed by a nonlinear ReLU layer with negative slope 0.2 except for Dec0 after which a tanh layer is used for the output. We implemented the architecture in caffe [3] framework. While training the network, we present the RGB-XYZ features of an image as input and use the ground truth object coordinates for computing the loss. The network is trained by minimizing a robust and smooth Huber loss function [2] using the Adam solver with a momentum of 0.9 and a learning rate of 1e-5.

We generate the ground truth object coordinates for the foreground cars using the following process. We obtain the car CAD model and the corresponding 6D pose annotations from Menze et al. [4]. Using the CAD model and the 6D pose, we render dense 3D object coordinates for each visible point on the surface of the car.

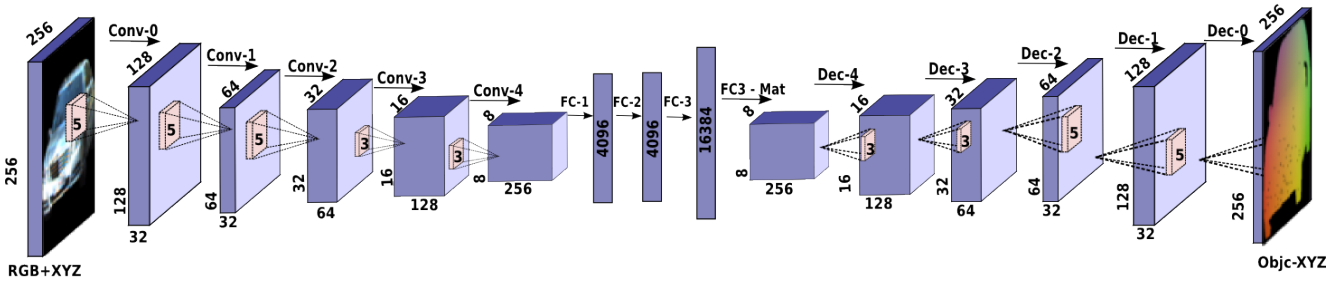

Figure 57: **Architecture of our Encoder Decoder Network for Object Coordinate Prediction.**

5. Random Forests vs. CNNs for Predicting Object Coordinates

In this section, we compare our object coordinate predictions to those of an existing state-of-the-art approach using random forests as presented by Brachmann et al. [1]. For this comparison, we use the same random forest (RF) implementation provided by [1]. In particular, their implementation estimates the pixel-wise object coordinates using an RGB-Depth patch of size 20x20 at each pixel of the instance. We train a random forest with 3 trees and maximum depth of 64 by sampling 3 million random points during training.

We remark that in [1], object coordinates are predicted only for a fixed set of known 3D objects. On the contrary, our case is a lot more complex due to the large intra-class variation among cars (hatchback, station wagon, SUV etc.). This presents a challenge to the prediction model as it has to generalize well for all of the possible car types. We found that our CNN based model generalized better to various types of cars and is more accurate in predicting object coordinates compared the state-of-the-art RF based approach. The average euclidean error of the predicted object coordinates using our CNN is 0.6 meters while random forests acchieve an average error of 2.89 meters. We believe that with more training data the performance of our object coordinate predictions could be further improved.

In Fig. 58, we compare the object coordinate predictions from our CNN to those of the RF. For illustration purposes, we have colorized the object coordinate ground truth. As clearly visible, the continuous smooth change of colours indicate a smooth fine grained part labelling of the car. In addition, we show the corresponding error map. In contrast to our predictions, the object coordinate predictions from the RF are extremely noisy and thus lead to very large errors. We conclude that random forests can't cope well with the appearance changes induced by intra-class variations present in our dataset.

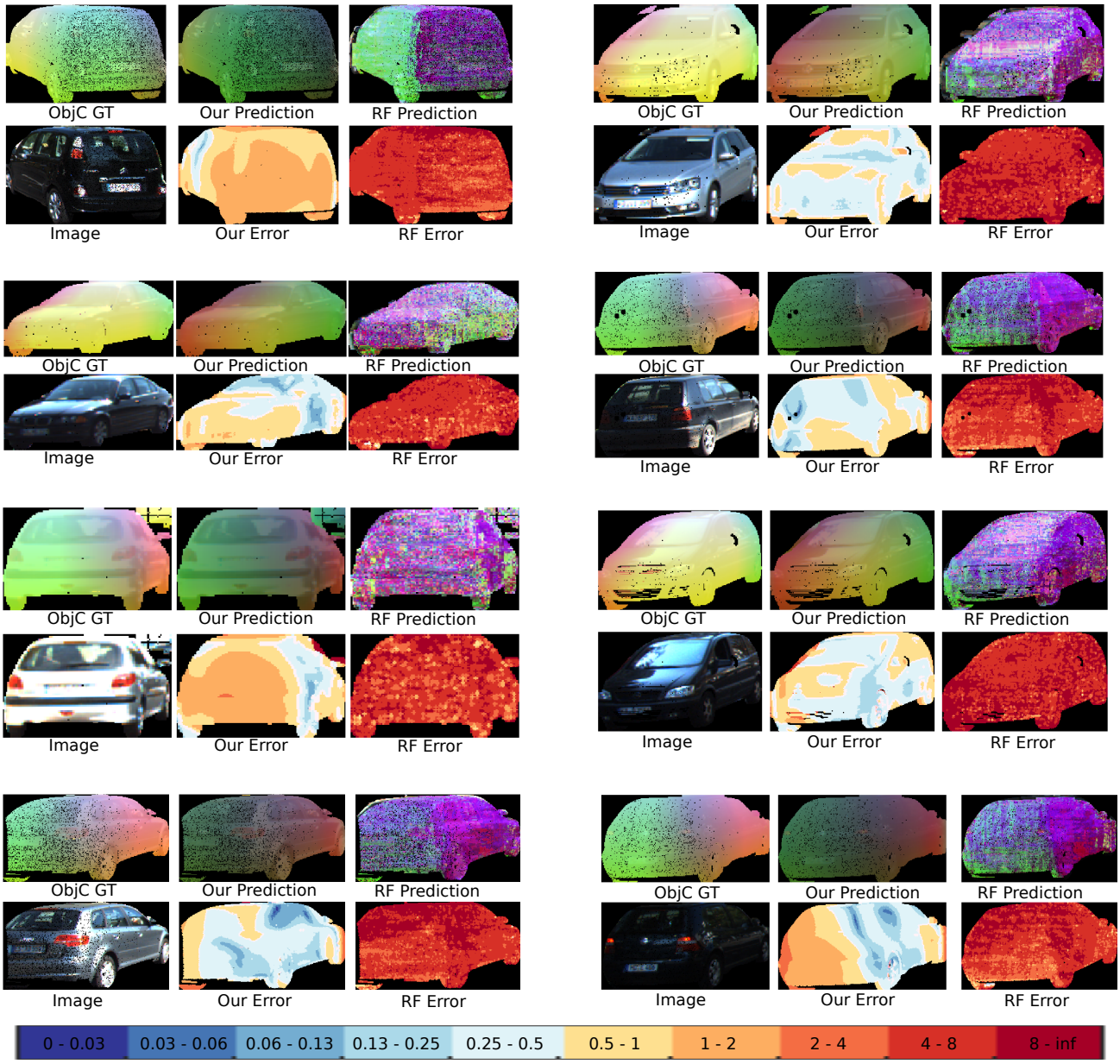

Figure 58: This figure shows a qualitative comparison of object coordinate predictions from our CNN and a baseline RF approach. In each of the illustrations, we show the ground truth object coordinates (ObjC GT), predictions from our CNN (Our Prediction), predictions from RF (RF Prediction), corresponding pixel wise error maps (Our Error, RF Error) and also the car instance for which the object coordinates are predicted. While the error of our CNN based object coordinate predictions is mostly below 1 meter, the RF based approach leads to much larger errors.

References

[1] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother. Learning 6d object pose estimation using 3d object coordinates. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2014. 32

[2] R. B. Girshick. Fast R-CNN. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2015. 32

[3] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 32

[4] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32

[5] M. Neoral and J. Šochman. Object scene flow with temporal consistency. In *Proc. of the Computer Vision Winter Workshop (CVWW)*, 2017. 23, 24, 25, 26, 27, 28, 29, 30, 31

[6] M. Tatarchenko, A. Dosovitskiy, and T. Brox. Multi-view 3d models from single images with a convolutional network. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2016. 32

[7] C. Vogel, K. Schindler, and S. Roth. 3d scene flow estimation with a piecewise rigid scene model. *International Journal of Computer Vision (IJCV)*, 115(1):1–28, 2015. 23, 24, 25, 26, 27, 28, 29, 30, 31