Consistency Guided Scene Flow Estimation – Supplementary Material –

In this supplementary material, we provide an analysis of the iterative refinement module.

The consistency-guided refinement module is used to iteratively refine the scene flow estimation. To gain further understanding of the refinement process, we provide an analysis on the intermediate results, *i.e.*, scene flow estimation after each refinement step.

Following the main paper, we use the same experimental setting in this study. In more details, the model is trained on the FlyingThings 3D dataset using ground-truth labels, and evaluated on the KITTI scene flow dataset. We evaluate the scene flow estimation results after each refinement step, using the four evaluation metrics: D1, D2, F1, SF, which represent the percentage of outliers. In the main paper, a total of 5 refinement steps are performed. Here we also report the performance of applying further refinement, up to 10 steps. We plot the results at different steps in fig A1. We observe that the error decreases notably in all four measurements, which shows that the refinement module is useful for improving scene flow accuracy. The improvement is more significant in the first 5 steps, then the performance saturates with additional steps. Further refinement leads to an increase in the overall running time, as per refinement step requires approximately 0.1 second.

We present some pixel-wise error visualizations in fig A2, where we show the error maps at different refinement steps. We can observe that the proposed refinement module also produces a notable qualitative improvement in every iteration, this further verifies the effectiveness of the consistency-guided refinement.

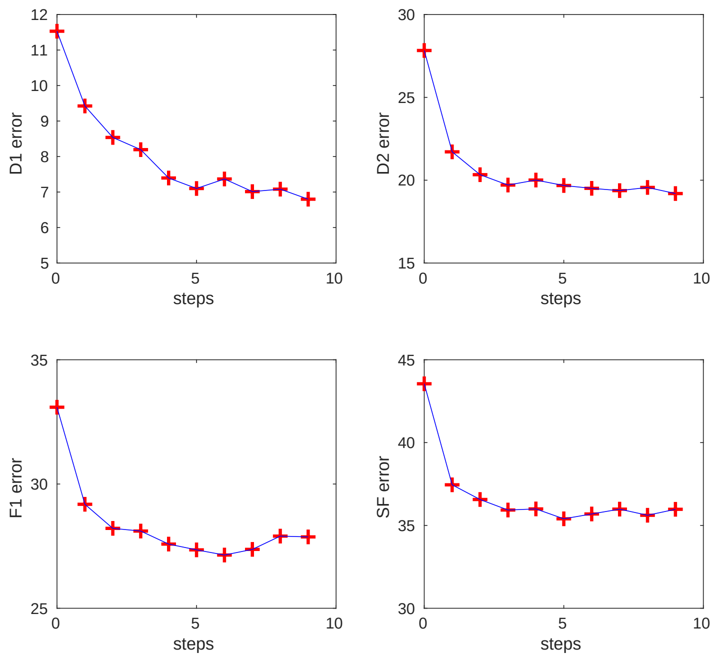

Fig. A1. The e ect of re nement steps on scene ow performance. We plot the error using D1, D2, F1 and SF measure in the above four gures, respectively. x-axis is the number of re nement steps, with 0 being the feedforward results(without the re nement module). y-axis is the percentage of outliers. A pixel is considered correct if the prediction end-point error is smaller than $3px$ or 5%. For SF, a correct pixel needs to be correct for D1, D2 and F1. The model is trained on FlyingThings3D, and test on KITTI scene ow dataset.