DFNet: Enhance Absolute Pose Regression with Direct Feature Matching Supplementary

Shuai Chen, Xinghui Li, Zirui Wang, and Victor A. Prisacariu

Active Vision Lab, University of Oxford

1 Supplementary

1.1 Implementation

Histogram-assisted NeRF Here, we provide more implementation details of our methods. As we mentioned in Section 3.3 of the main paper, our histogram-assisted NeRF model renders at a speed of 12.2 fps (benchmarked by a 3080Ti GPU) to achieve online RVS training. In order to achieve a balanced trade-off between speed and quality, we choose to render small images with a shorter side of 60 pixels. In addition, we set the NeRF model architecture to 64 coarse and 64 fine sampling with an MLP width of 128.

DFNet Our DFNet takes an image input with a shorter side of 240 pixels. For feature extractor module $\mathcal{G}$ of DFNet, features are fed through a Conv-Relu-Conv-Batch Norm architecture with 64 kernels and 128 output channels. The DFNet is trained with a batch size of 4 or 8, depending on the GPU's memory. We implement an early stopping strategy with a patient value of 200 and schedule the learning rate decay of 0.95 when validation loss plateaus for every 50 epochs. For every $N = 20$ epochs, we will randomly generate the same amount of views as the training sample size using RVS.

Direct Feature Matching To train the DFNet$_{dm}$ model with feature-metric direct matching using unlabeled data, we set the batch size to 1 and the learning rate to 1×10^{-5} with the same early stopping strategy mentioned above. We discover that only low-level features (i.e., features from the first blocks of VGG) are needed to achieve the best performance, which we discussed earlier in Section 4.5 of the main paper.

1.2 Visualization

Qualitative Comparison on 7-Scenes We show a selection of qualitative comparisons on the 7-Scenes dataset with several baseline APR methods [4,1,2] in Fig. 1. We also encourage our readers to check out our supplementary video, in which we rendered views of the predicted pose using NeRF synthesis.

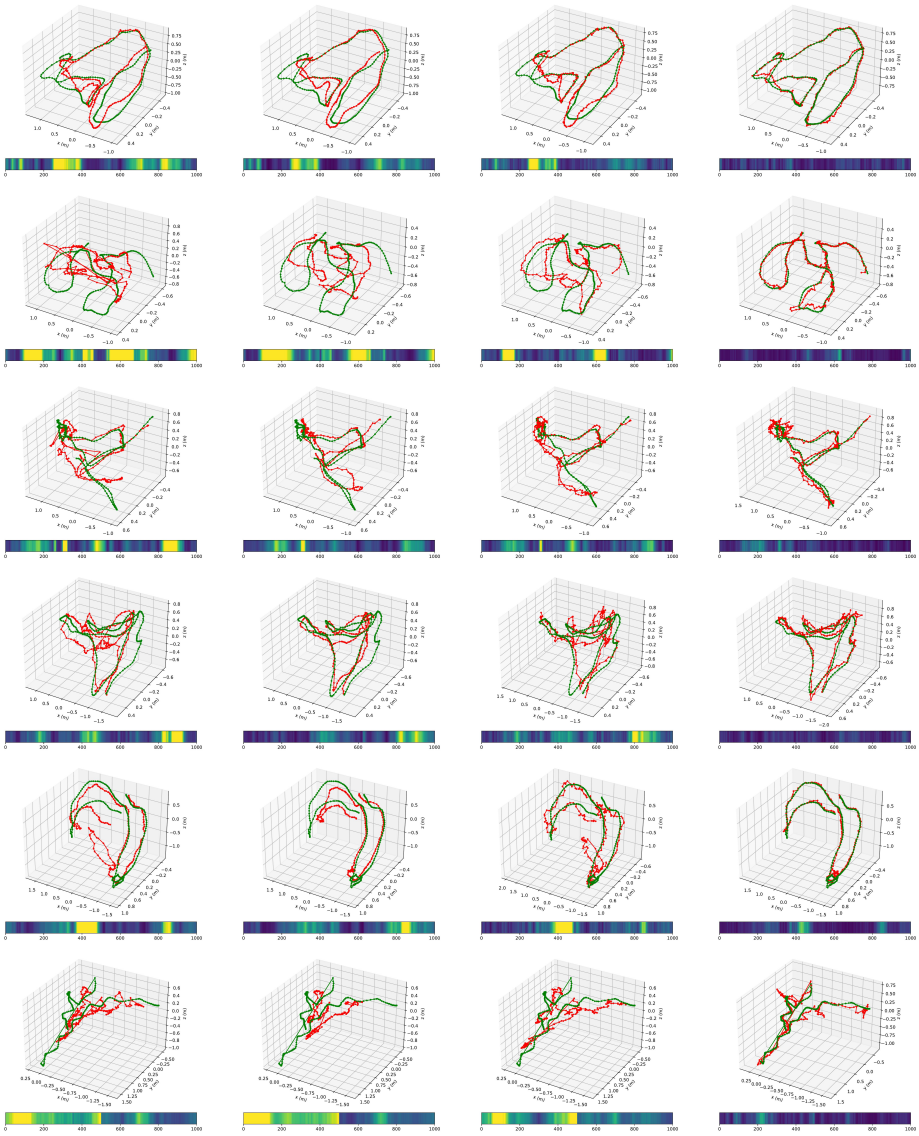
NeRF-W vs. Histogram-assisted NeRF In real-life camera localization applications, since training and testing data are likely to be taken from different sequences, camera exposures, or time of the day, our histogram-assisted NeRF would be more helpful to render accurate appearcance Fig. 2. We experimentally found our histogram-assisted NeRF is helpful in both photometric matching and feature-metric matching approaches.

1.3 Additional Discussion

Photometric Distortion As discussed in section 3.2 of the main paper, photometric matching relies on RGB-wise differences between the query and rendered images. However, if those images appear in different lighting/exposure conditions, the photometric loss will fail due to large RGB-wise differences even under the same camera pose. Previous photometric matching approaches, such as Direct-PN+U, perform worse in pose estimation when using unlabeled data with large appearance variations from the training sequences (refer to the lower part of main paper Table 4b). We observed that such degradation is consistent in other outdoor scenes.

Two Properties of Domain Invariant Features We had two clear goals for designing our robust feature extractor: (1) We want the extracted features to be sensitive to pose changes. (2) We want the features to be indistinguishable between real and rendered image features from the same pose (Close the Domain Gap). Our first goal is achieved by the L_2 pose loss supervision, which ensures the features are closely related to the pose regression task. We specifically design the Feature Extractor to share the backbone with the Pose Module (see main paper Fig 2a). Although the deeper layer features may lose semantic meaning, we observe that those features can respond to pose changes.

The triplet loss is primarily designed to achieve the second goal without feature collapse in the training process. We previously tried to force real and rendered image features to be the same by using MSE/L_2 losses, leading to feature collapse (main paper Fig 6a). This is because the pink layers in main paper Fig 2a, despite being shallow, are likely to learn to cheat since those layers are not supervised by other meaningful losses. Thus, we introduce the triplet loss to prevent features collapsing. We experimentally find that the proposed in-triplet mining adds extra robustness to both feature extraction and pose regression and leads to better APR performance overall. Such observation could hint that removing the domain gap benefits APR training when using extra randomly generated synthetic training data.



(a) PoseNet[5,3,4] (b) MapNet+PGO[1] (c) Direct-PN+U[2] (d) DFNet$_{dm}$

Fig. 1. Qualitative comparison on the 7-Scenes dataset. The 3D plots show the camera positions, green for ground truth and red for predictions. The bottom color bar represents rotational errors for each subplot, where yellow means large error and blue means small error for each test sequence. Sequence names from top to bottom are: Chess-seq-03, Fire-seq-04, Office-seq-07, Kitchen-seq-06, Kitchen-seq-12, Stairs-all.

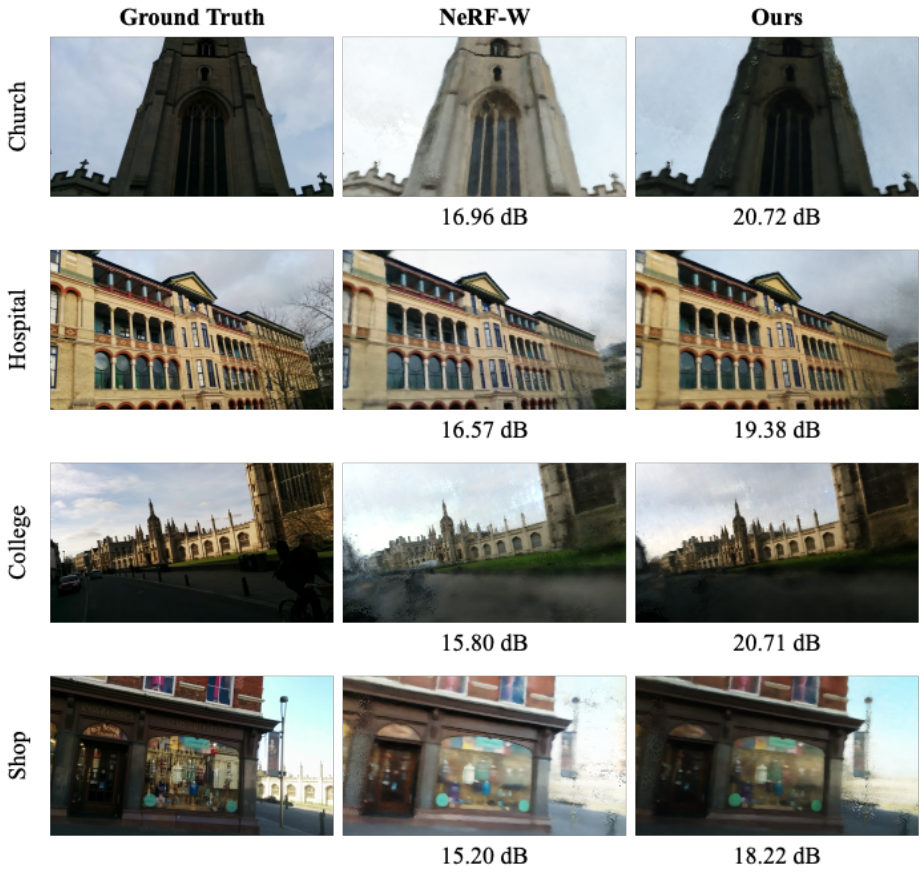


Fig. 2. A visual comparison between NeRF-W and our histogram-assisted NeRF on the testing sequences of Cambridge Landmarks dataset. The corresponding scene's test PSNR is displayed at the bottom of each sub-figure.

References

1. Brahmbhatt, S., Gu, J., Kim, K., Hays, J., Kautz, J.: Geometry-Aware Learning of Maps for Camera Localization. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
2. Chen, S., Wang, Z., Prisacariu, V.: Direct-PoseNet: Absolute pose regression with photometric consistency. In: 3DV (2021)
3. Kendall, A., Cipolla, R.: Modelling uncertainty in deep learning for camera relocalization. In: ICRA (2016)
4. Kendall, A., Cipolla, R.: Geometric loss functions for camera pose regression with deep learning. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
5. Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-dof camera relocalization. In: International Conference on Computer Vision (2015)