Appendix for paper "NCTR: Neighborhood Consensus Transformer for Local Feature Matching" submitted to ICIP2022.

A Experimental details

A.1. Homography estimation

The test set contains 500 images which are resized to 640×480. The groundtruth homographies are generated by applying random perspective, scaling, rotation, and translation to test images. We detect 512 keypoints on each image with NMS radius of 4 pixels and keypoint threshold of 0.005. We generate match with match threshold of 0.44 and Sinkhorn iterations of 20. A match is considered correct if the reprojection error of the match is less than 3 pixels. Two methods are applied to calculate homography. The basic method is *cv2.findHomography* with 3000 iterations and a RANSAC inlier threshold of 3 pixels. The improved method is *pydegensac.findHomography* with the default setting of DEGENSAC.

A.2. Outdoor pose estimation

Test images are resized to 640×480 for processing speed and memory usage. We detect 2048 keypoints on each image with NMS radius of 3 pixels and keypoint threshold of 0.005. We generate match with match threshold of 0.22 and Sinkhorn iterations of 20. A match is considered correct if the epipolar distance is less than 1e-4. Poses are computed by applying OpenCV's *findEssentialMat* with an inlier threshold of 1 pixel divided by the focal length and *recoverPose* function.

B Additional Experiment Results

B.1 Homography estimation

Figure 2 shows some visualization results of the homography estimation experiment, the color of the line represents the matching confidence. In the first three rows, there are matches in the NN and SuperGlue results that do not conform to neighborhood consensus, which means that adjacent points in the same image are matched to very different regions in another image. This problem is solved by NCTR through incorporating neighborhood consensus. And it can be seen from the last two rows that NCTR is able to get more matches with higher confidence than other methods. As shown in Figure 1, NCTR outperforms SuperGlue at most thresholds, guaranteeing both the number of matches and the matching precision.

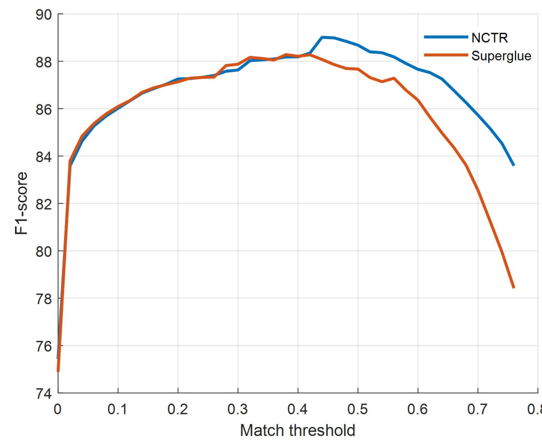


Figure 1: F1-score of NCTR and Superglue at different match thresholds.

B.2 Outdoor pose estimation

Figure 4 shows that compared with SuperGlue, which also applies the attention mechanism, incorporating neighborhood consensus information into descriptors results in significantly fewer false matches, bringing precision of NCTR to state-of-the-art performance. As shown in Figure 3, NCTR outperforms SuperGlue at most thresholds in outdoor pose estimation, NC module is able to improve precision while maintaining a large number of matches, building a better foundation for upstream tasks, such as 3D reconstruction and image retrieval.

B.3. Efficiency analysis

In addition to the validation of model performance, the efficiency of the model is also critical. Based on a 2060S GPU, NCTR is compared against the handcrafted method Nearest Neighbor and the learned method SuperGlue. We set the image crop size to 640×480 and the number of keypoints to 512, and compute the average runtime of these methods on the test set. The NN method runs at 6.07 FPS, SuperGlue runs at 5.42 FPS and NCTR runs at 5.04 FPS. Incorporating the NC module slightly increases runtime, but doesn't destroy real-time performance for applications. We noticed that the SuperPoint network took up some time in the total processing time. Because of the high robustness of our method, in practical applications, the image cropping size and number of keypoints can be adjusted according to the actual situation to reduce the SuperPoint processing time.

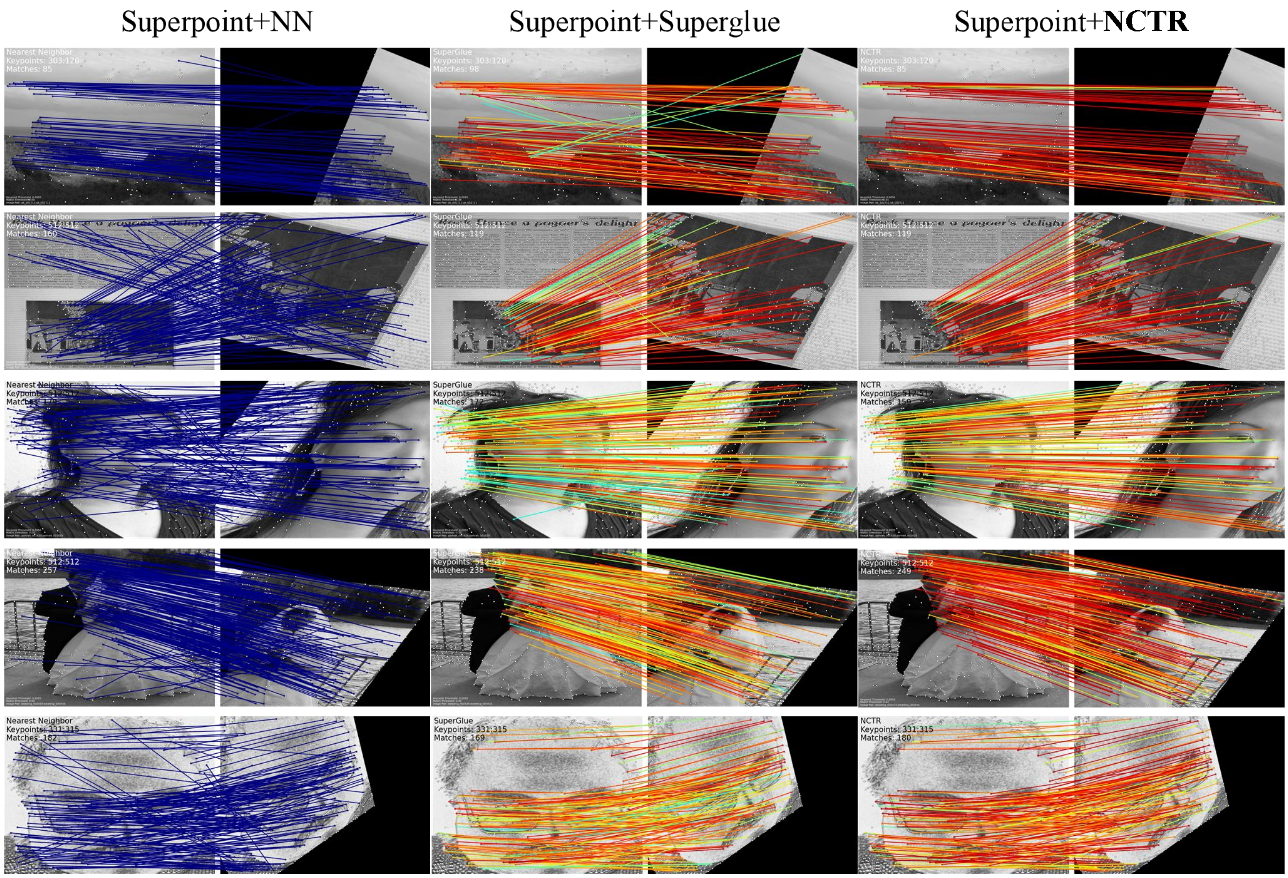
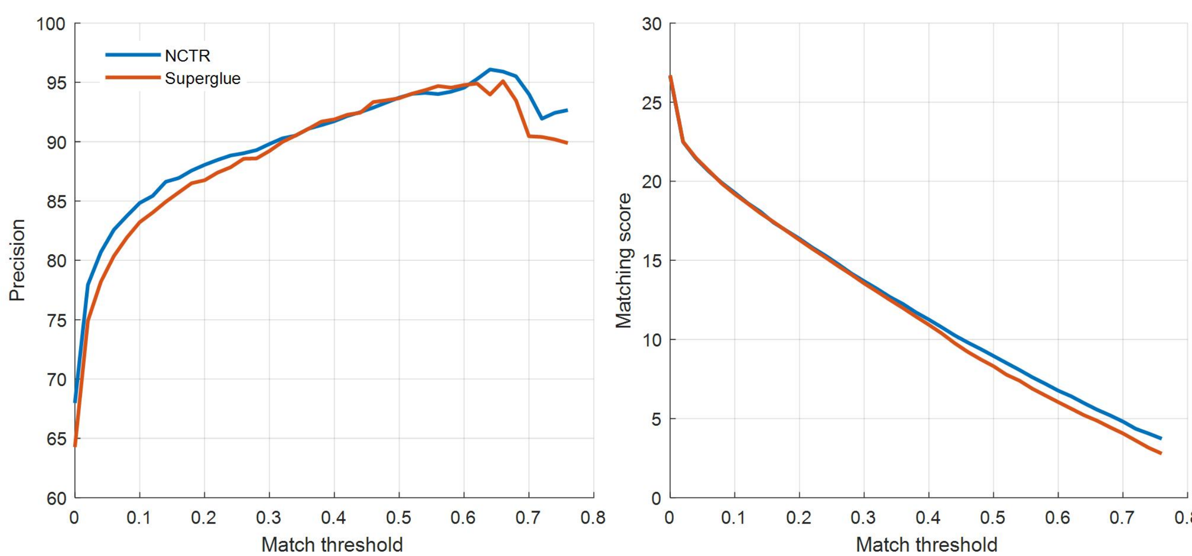


Figure 2: Visualization results of homography estimation. The matching confidence is from low to high.



(a) (b)

Figure 3: Precision (a) and MS (b) of NCTR and SuperGlue at different match thresholds.

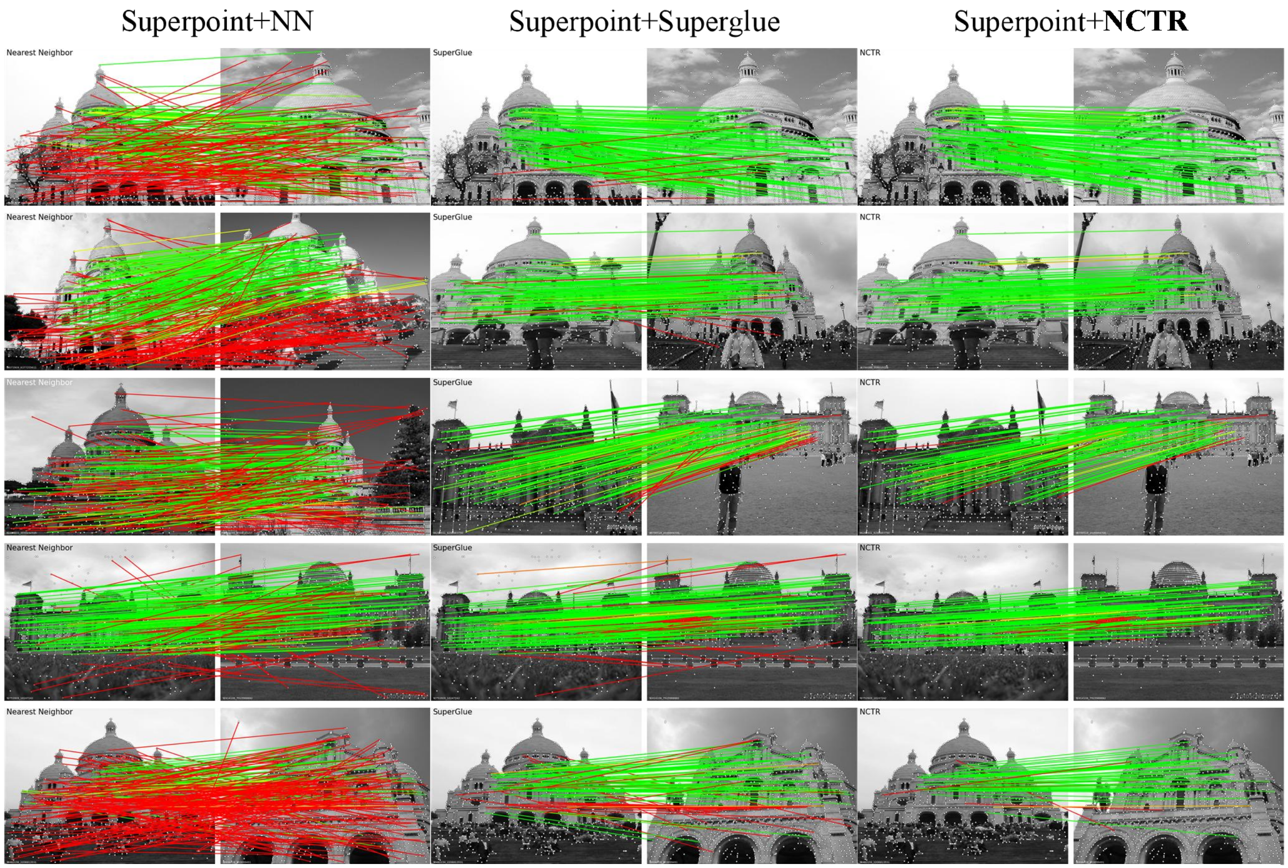


Figure 4: Visualization results of pose estimation. Correct and false matches are shown in green and red.