

# Neural Attention Field: Emerging Point Relevance in 3D Scenes for One-Shot Dexterous Grasping

## – Appendix –

Anonymous Author(s)

Affiliation

Address

email

### 1 A Network Details

2 **Transformer decoder architecture** Fig. A illustrates the network architecture of our transformer  
3 decoder. We compute the pairwise distances between the hand points and scene points and apply  
4 them as weights to the keys and queries for more stable convergence during training.

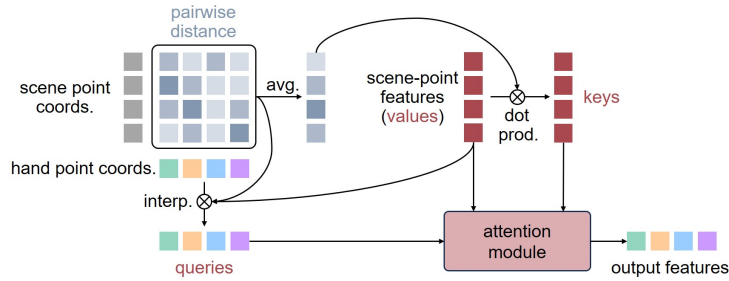


Figure A: Network architecture of the transformer decoder.

5 **Network training** The transformer decoder network is only trained at the feature pre-training  
6 stage with the self-consistency losses, and all network weights are frozen afterward in the end-  
7 effector optimization stage. All networks are trained on a single GeForce RTX 3090 with an Adam  
8 optimizer for 100 iterations.

### 9 B Real-Robot Setup Details

10 Fig. B shows our real-robot setup. Four Femto Bolt  
11 sensors are mounted at the four corners 100cm above  
12 the table, with each edge 100cm. The objects  
13 are placed near the center of the table. The dexterous  
14 hand is mounted on a UR10e arm, reaching the ob-  
15 jects from the left.

16 **Shadow hand parameterization** As mentioned in  
17 the paper, we conduct all our experiments in the real  
18 world with a Shadow Dexterous Hand of 24 DoF.  
19 We restrict the 2 DoF at the wrist, focusing the opti-  
20 mization on the remaining 22 DoF. The UR10e arm  
21 also introduces 6 additional DoFs. Each hand joint

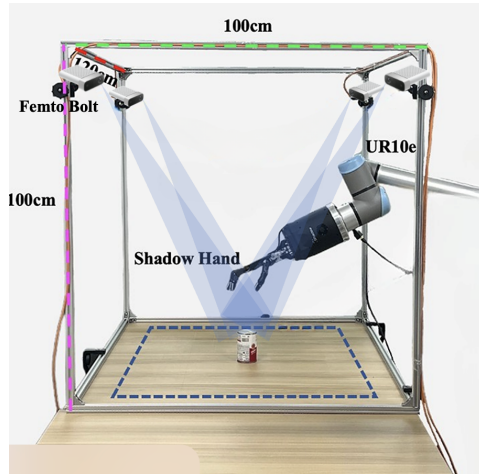


Figure B: Real-robot setup.

is parameterized as a scaler for its rotation angle. The shadow hand has 4 under-actuated joints. But in our method, we just regard it as a normal actuated joint, which is a common practice and the default API from the shadow hand takes these four joints as normal joints. We use “rotate6D” to represent the 3-DoF hand base rotation, which is the first two columns of the rotation matrix.

**Scene setup and randomization** In the real-robot experiments, for single-object scenes with toy animals, we randomize its pose with arbitrary z-axis rotations so that the dexterous hand can grasp it from the top without hitting the table. For functional grasping with 3D-printed tools, we put them on boxes so that the robot can reach its handle without hitting the table. For multi-object scenes, we randomize the poses of all objects with full  $SO(3)$  rotations.

## C Ablation Study Details

**Evaluation metrics (Fig. 6)** The ablation studies are done virtually on pre-captured 3D scene pointclouds to eliminate the variances caused by scene randomization, and thus the 10 evaluation scenes in all the ablation settings are identical. In Fig. 6, we evaluate the average distance between the hand-surface query points and the target region (monkey arm) in the scene, as a rough metric indicating how close the hand reaches the target post-optimization. The target region is manually annotated with MeshLab.

**Quantitative results** Fig. C shows the qualitative results on grasping the monkey arm with the same demonstration but different pre-training setups for the transformer decoder. The hand optimization trajectories are shown in green.

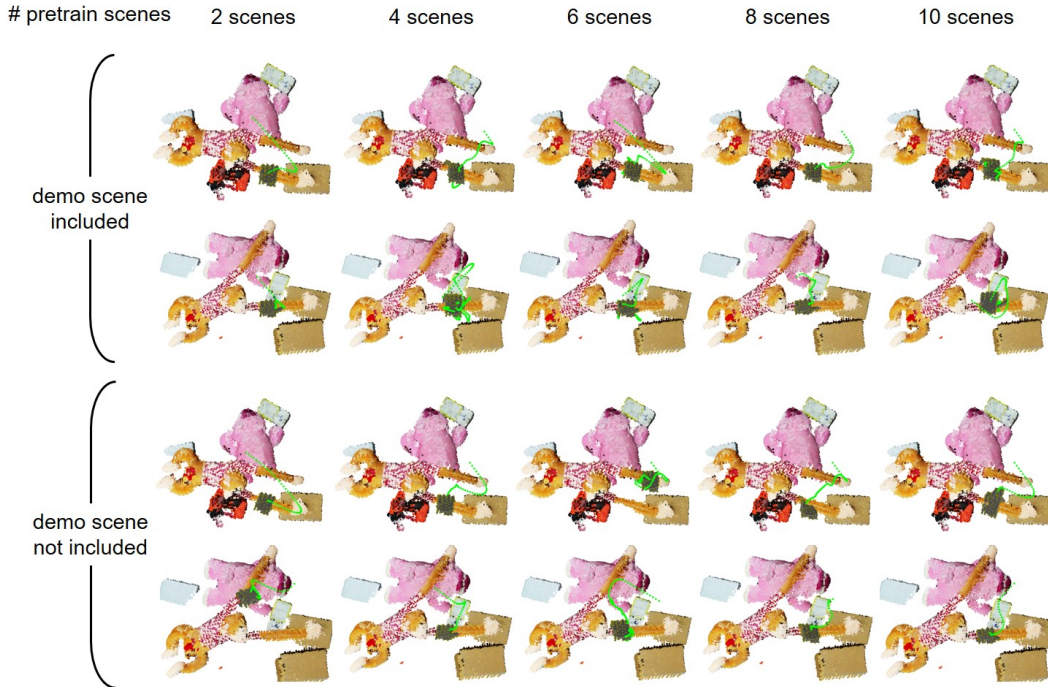


Figure C: Qualitative results for the ablation study on different pre-training setups.