TetraSphere: A Neural Descriptor for O(3)-Invariant Point Cloud Analysis

Supplementary Material

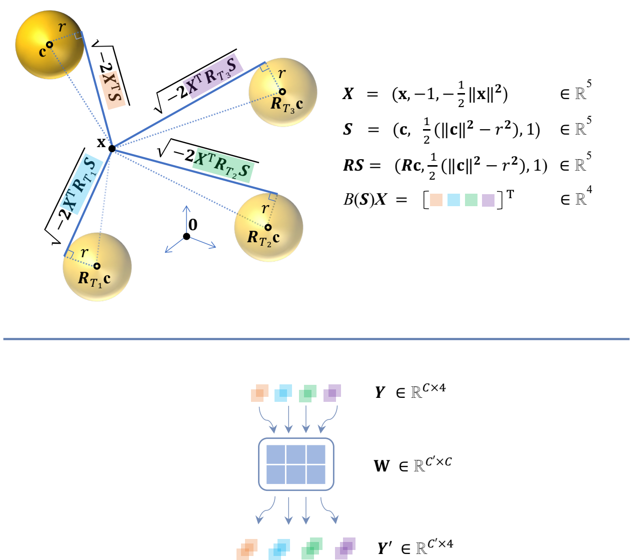

Figure 4. (Best viewed in color.) **Top**: Tetra-basis projection is the output of a steerable 3D spherical neuron [28] Without loss of generality, consider one ($K = 1$) steerable spherical neuron $B(\boldsymbol{S})$ (see Section 3.3) with $\boldsymbol{R}_O = \mathbf{I}_5$, and the input point $\mathbf{x}$ that happens to lie outside of the sphere $(\mathbf{c}, r)$ with the learnable parameter vector $\boldsymbol{S}$ (assume $\gamma = 1$, and thus $\tilde{\boldsymbol{S}} = \boldsymbol{S}$; see Section 3.2) and its three rotated copies. Then the projection of $\mathbf{x}$ in the tetra-basis $B(\boldsymbol{S})$ is the vector $B(\boldsymbol{S})\boldsymbol{X}$ consisting of four scalar activations $\boldsymbol{X}^\top \boldsymbol{R}_{T_i}\boldsymbol{S}$ of the respective spherical decision surfaces. Each activation determines the respective cathetus length, as per [27]. **Bottom**: Vector neurons [10] preserve the spatial dimension (4 in our case) and alter the latent dimension C of the feature $\boldsymbol{Y}$, see (10).

7. Additional illustrations

In order to help the reader to understand the main concepts of our approach, *i.e.*, prior work (steerable) spherical neurons [28] and vector neurons [10], as well as 4D tetra-basis projections (see Figure 1 and Section 4.1), we provide illustrations in Figure 4.

8. Learned Tetra-selection

In this section, we present the Tetra-selection discussed in Section 5.3. As we can see from Figures 5 and 6, **TetraSphere** learns all but one γ parameter of the spherical decision surface (see (5)), defining the steerable neuron (6), to be close to 0, effectively always selecting one tetra-basis (out of K) during inference. We attribute the increased performance for $K > 1$ (see Tables 1, 2, and 3), to the higher chance of selecting a better initialization of the steerable neuron parameters.

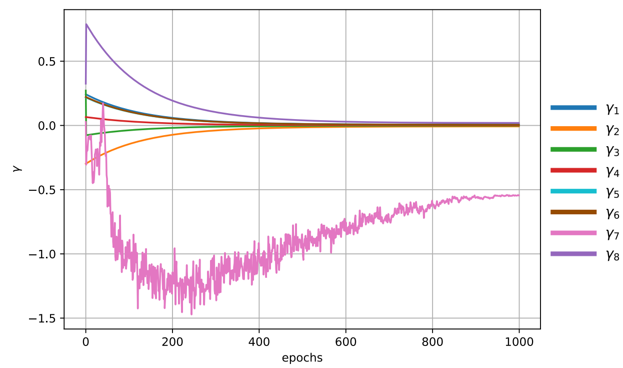

Figure 5. Learned γ parameters for **TetraSphere**$_{K=8}$ trained on the *OBJ_BG* subset of ScanObjectNN (see Table 1). All but γ_7 converge close to 0.

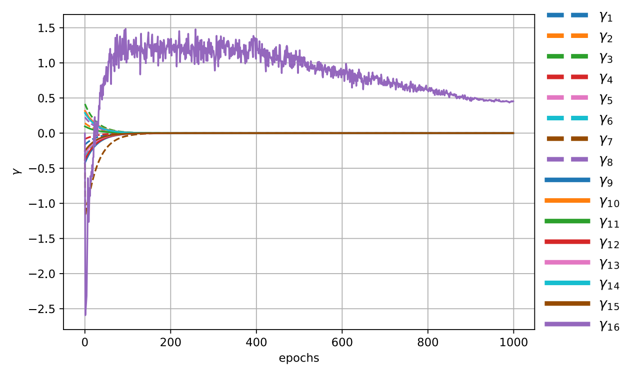

Figure 6. Learned γ parameters for **TetraSphere**$_{K=16}$ trained on the *PB_T50_RS* (see Table 2) of ScanObjectNN. All but γ_{16} converge close to 0.

9. Synthetic data results

We present a complete comparison of the methods trained on synthetic data to perform classification and part segmentation in Tables 5 and 6, respectively. Our **TetraSphere** achieves the best performance among equivariant methods in both tasks, consistently outperforming VN-DGCNN.

Only the two RI methods PaRINet [6] and Yu *et al.* [48] outperform tetrasphere in the former case and only PaRINet in the latter. Note that TetraSphere outperforms both PaRINet and Yu *et al.* on the other two real-data benchmarks (see Tables 1 and 2).

Methods	z/z	$z/$ SO(3)	SO(3)/ SO(3)
	Rotation-sensitive		
PointCNN [25]	**92.5**	41.2	84.5
DGCNN [41]	90.3	33.8	88.6
	Rotation-invariant		
3D-GFE [8]	88.6	89.4	89.0
Li *et al.* [23]	90.2	90.2	90.2
Yu *et al.* [48]	91.0	<u>91.0</u>	<u>91.0</u>
PaRINet [6]	<u>91.4</u>	**91.4**	**91.4**
	Rotation-equivariant		
TFN [31]	89.7	89.7	89.7
VN-DGCNN [10]	89.5	89.5	90.2
TetraSphere$_{K=1}$	89.5	89.5	89.9
TetraSphere$_{K=2}$	89.7	89.7	90.0
TetraSphere$_{K=4}$	90.0	90.0	89.5
TetraSphere$_{K=8}$	90.5	90.5	90.3
TetraSphere$_{K=16}$	89.8	89.8	90.0

Table 5. Classification acc. (%) on the ModelNet40 shapes under different train/test settings of rotation augmentation. The overall best results are presented in **bold**, and the second best are <u>underlined</u>. Our **TetraSphere** sets a new state-of-the-art performance for equivariant baselines.

Methods	z/z	$z/$ SO(3)	SO(3)/ SO(3)
	Rotation-sensitive		
PointCNN [25]	**84.6**	34.7	71.4
DGCNN [41]	82.3	37.4	73.3
	Rotation-invariant		
3D-GFE [8]	-	78.2	77.7
Li *et al.* [23]	81.7	81.7	81.7
PaRINet [6]	<u>83.8</u>	**83.8**	**83.8**
Yu *et al.* [48]	-	80.3	80.4
	Rotation-equivariant		
TFN [31]	-	78.1	78.2
VN-DGCNN [10]	81.4	81.4	81.4
TetraSphere$_{K=1}$	82.1	82.1	82.3
TetraSphere$_{K=2}$	82.3	82.3	82.5
TetraSphere$_{K=4}$	82.2	82.2	82.2
TetraSphere$_{K=8}$	82.3	82.3	82.4
TetraSphere$_{K=16}$	82.3	82.3	82.3

Table 6. Part segmentation: ShapeNet mIoU (%). The overall best results are presented in **bold**, and the second best are <u>underlined</u>. Our **TetraSphere** sets a new state-of-the-art performance for equivariant baselines.