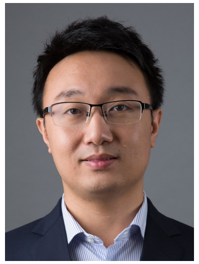
Interpreting Representation Quality of DNNs for 3D Point Cloud Processing



Wen Shen[b,*], Qihan Ren[a], Dongrui Liu[a], Quanshi Zhang[a,†]

[a]Shanghai Jiao Tong University, Shanghai, China
[b]Tongji University, Shanghai, China
*This work was done when Wen Shen was an intern at Shanghai Jiao Tong University.
†Correspondence

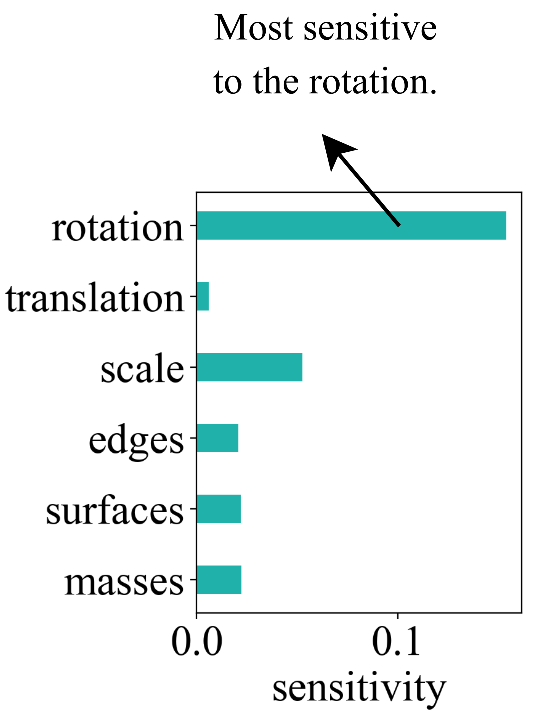
Objective

Analyze the **quality of representation** of DNNs for 3D point cloud processing

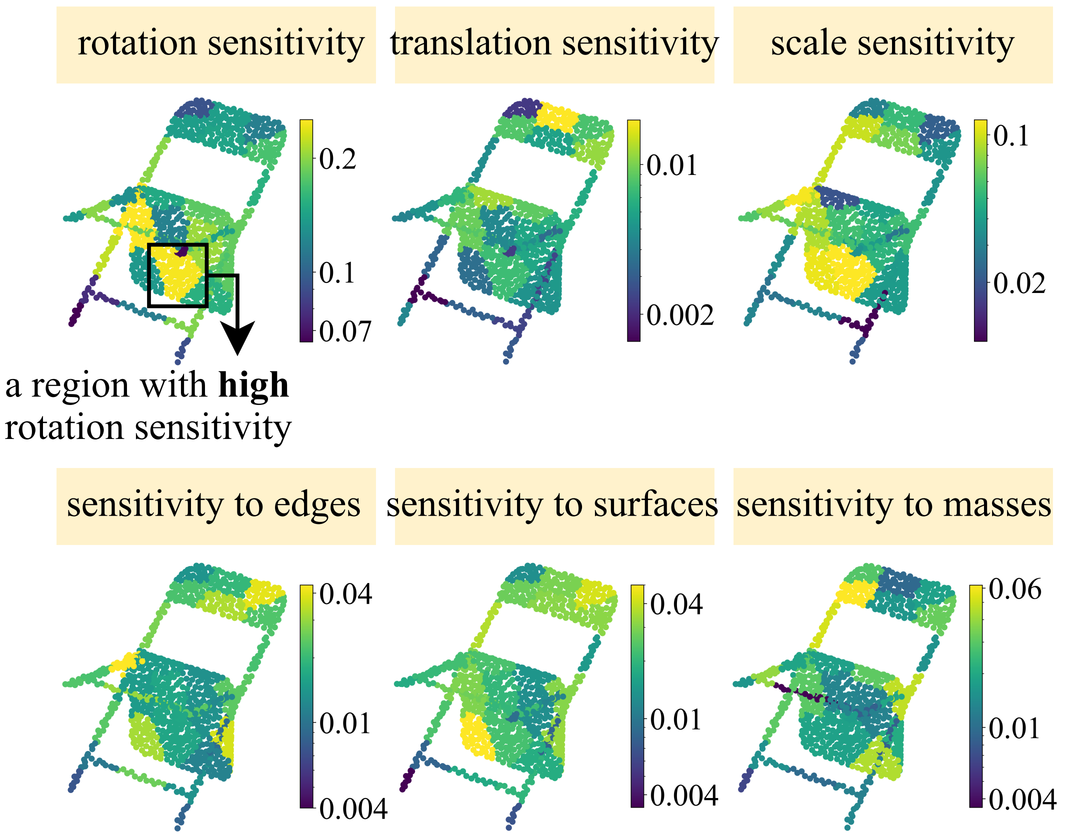
- Regional sensitivities
- Spatial smoothness
- Representation complexity

Introduction

Regional sensitivities



(a) Comparison of six types of sensitivities of PointNet++.



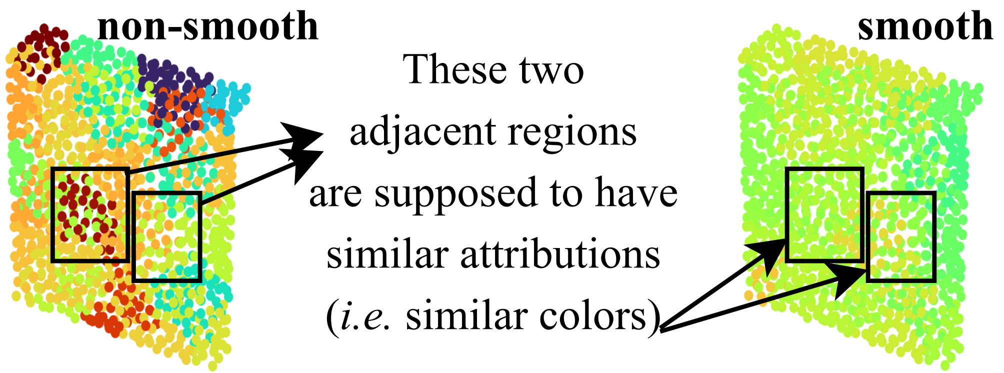
(b) Visualization of regional sensitivities of PointNet++.

Disentangle the overall model vulnerability into six types of regional sensitivities.

Introduction

Spatial smoothness

Adjacent regions are supposed to have similar attributions to the network output.

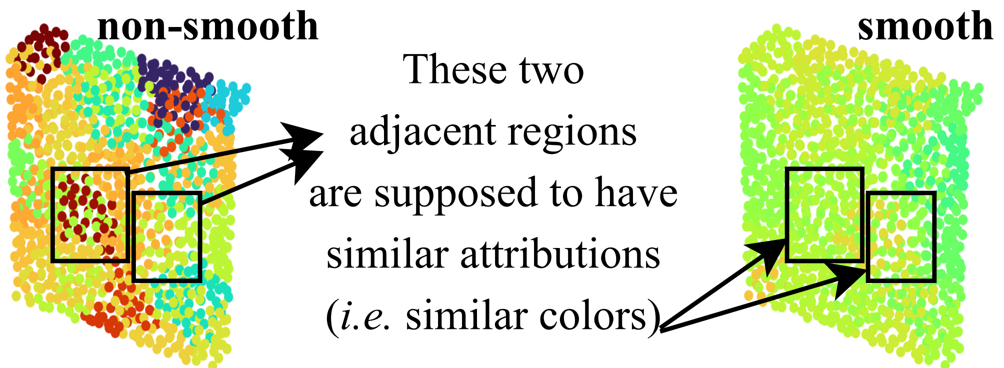


(c) The smoothness of regional attributions.

Introduction

Spatial smoothness

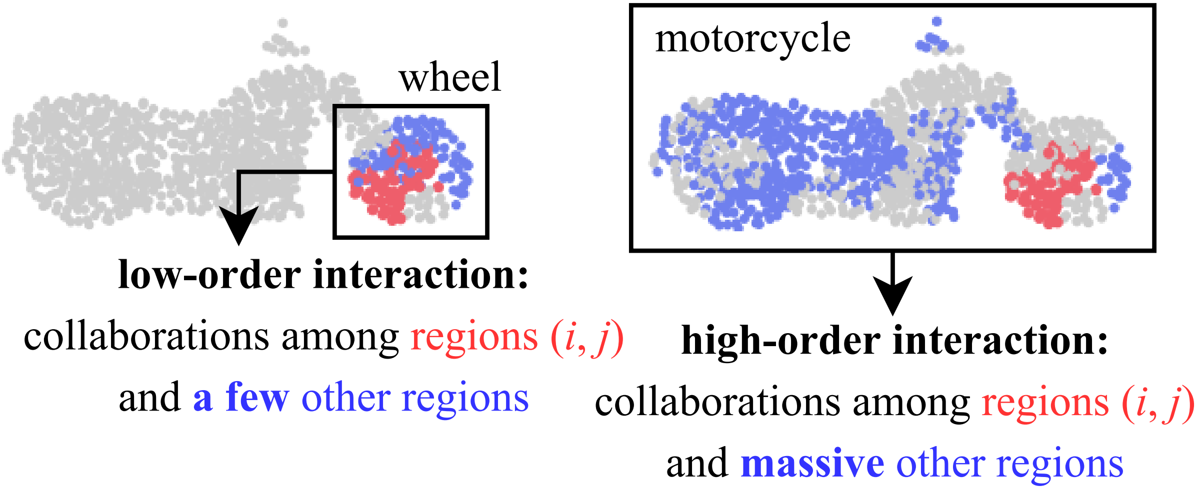
Adjacent regions are supposed to have similar attributions to the network output.



(c) The smoothness of regional attributions.

Representation complexity

Evaluate the complexity of 3D structures that can be encoded in a DNN.

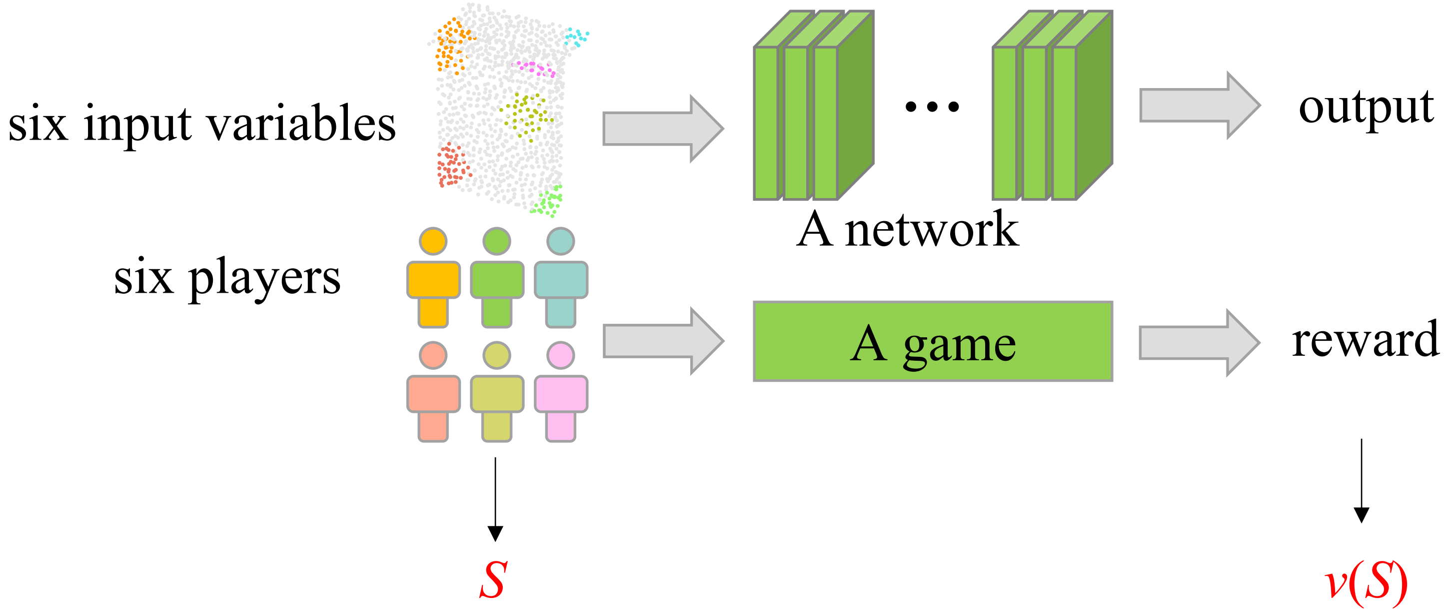


(d) Illustration of the multi-order interaction.

Preliminaries: Shapley values

- A unique unbiased approach to fairly allocate the total reward to each player[1]
- Satisfies axioms of *linearity*, *nullity*, *symmetry*, and *efficiency*[2]

In 3D point cloud processing → Game
- Input point cloud regions → players
- Scalar output of the DNN → total reward of the players in the game



[1]Lloyd S Shapley. A value for n-person games. Contributions to the Theory of Games, 2(28):307–317, 1953.
[2] Robert J Weber. Probabilistic values for games. The Shapley Value. Essays in Honor of Lloyd S. Shapley, pages 101–119, 1988.

Preliminaries: Shapley values

The numerical attribution of the i-th region can be estimated by the Shapley value $\phi(i)$.

$$\phi(i) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(n - |S| - 1)!}{n!} (v(S \cup \{i\}) - v(S))$$

- $N = \{1, 2, \dots, n\}$ denotes input variables (point cloud regions)
- $v(S) = \log \frac{p}{1-p}$, where $p = p(y = y^{\text{truth}} | x_S)$
- x_S denotes the point cloud only containing regions in $S \subseteq N$

Regional sensitivities

The rotation/translation/scale/local-structure sensitivity of region *i* is quantified as the range of changes of this region's attribution $\phi(i)$ among all potential transformations $\{T\}$ of the rotation/translation/scale/local 3D structure.

$$\forall\, i \in N = \{1, 2, \cdots, n\}, \quad a_i(x) = \frac{1}{\boxed{Z}}\left(\max_T \phi_{x'=T(x)}(i) - \min_T \phi_{x'=T(x)}(i)\right)$$

$Z = \mathbb{E}_T[\sum_{i\in N} |\phi_{x'=T(x)}(i)|]$ is computed for normalization.

$$\boxed{\mathit{sensitivity} = \mathbb{E}_{x\in X}\left[\mathbb{E}_{i\in N}[a_i(x)]\right]}$$

Regional sensitivities

$$\forall\, i \in N = \{1, 2, \cdots, n\}, \quad a_i(x) = \frac{1}{Z}(\max_T \phi_{x'=T(x)}(i) - \min_T \phi_{x'=T(x)}(i))$$

- **Rotation sensitivity:** enumerate all rotation angles $\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3]^\top$ from the range of $[-\frac{\pi}{4}, \frac{\pi}{4}]$, and obtain a set of rotated point clouds $\{x' = T_{\text{rotation}}(x|\boldsymbol{\theta})\}$.
- **Translation sensitivity:** enumerate all translations $\Delta x = [\Delta x_1, \Delta x_2, \Delta x_3]^\top$ from the range of $[-0.5, 0.5]$, and obtain a set of translated point clouds $\{x' = T_{\text{translation}}(x|\Delta x) = x + \Delta x\}$.
- **Scale sensitivity:** enumerate all scales α from the range of $[0.5, 2]$, and obtain a set of scaled point clouds $\{x' = T_{\text{scale}}(x|\alpha) = \alpha x\}$.

Regional sensitivities

$$\forall\, i \in N = \{1, 2, \cdots, n\}, \quad a_i(x) = \frac{1}{Z}\left(\max_T \phi_{x'=T(x)}(i) - \min_T \phi_{x'=T(x)}(i)\right)$$

- **Rotation sensitivity:** enumerate all rotation angles $\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3]^\top$ from the range of $[-\frac{\pi}{4}, \frac{\pi}{4}]$, and obtain a set of rotated point clouds $\{x' = T_{\text{rotation}}(x|\boldsymbol{\theta})\}$.
- **Translation sensitivity:** enumerate all translations $\Delta x = [\Delta x_1, \Delta x_2, \Delta x_3]^\top$ from the range of $[-0.5, 0.5]$, and obtain a set of translated point clouds $\{x' = T_{\text{translation}}(x|\Delta x) = x + \Delta x\}$.
- **Scale sensitivity:** enumerate all scales α from the range of $[0.5, 2]$, and obtain a set of scaled point clouds $\{x' = T_{\text{scale}}(x|\alpha) = \alpha x\}$.

Regional sensitivities

$$\forall\, i \in N = \{1, 2, \cdots, n\}, \quad a_i(x) = \frac{1}{Z}(\max_T \phi_{x'=T(x)}(i) - \min_T \phi_{x'=T(x)}(i))$$

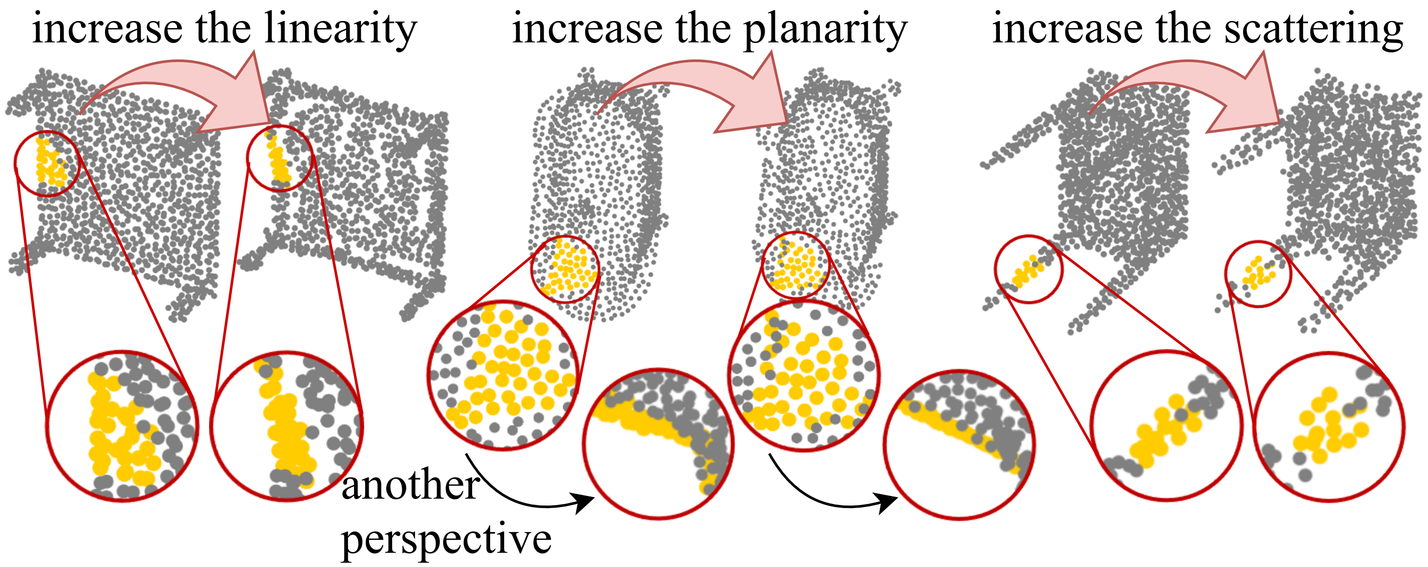
- **Rotation sensitivity:** enumerate all rotation angles $\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3]^\top$ from the range of $[-\frac{\pi}{4}, \frac{\pi}{4}]$, and obtain a set of rotated point clouds $\{x' = T_{\text{rotation}}(x|\boldsymbol{\theta})\}$.
- **Translation sensitivity:** enumerate all translations $\Delta x = [\Delta x_1, \Delta x_2, \Delta x_3]^\top$ from the range of $[-0.5,0.5]$, and obtain a set of translated point clouds $\{x' = T_{\text{translation}}(x|\Delta x) = x + \Delta x\}$.
- **Scale sensitivity:** enumerate all scales α from the range of $[0.5,2]$, and obtain a set of scaled point clouds $\{x' = T_{\text{scale}}(x|\alpha) = \alpha x\}$.

Regional sensitivities

- **Sensitivity to linearity (edge-like structures), sensitivity to planarity (surface-like structures), and sensitivity to scattering (mass-like structures).**

$$linearity = \frac{\lambda_1 - \lambda_2}{\lambda_1}; \quad planarity = \frac{\lambda_2 - \lambda_3}{\lambda_1}; \quad scattering = \frac{\lambda_3}{\lambda_1};$$

where $\lambda_1 \geq \lambda_2 \geq \lambda_3$ are eigenvalues of the covariance matrix of a region's points[1][2].



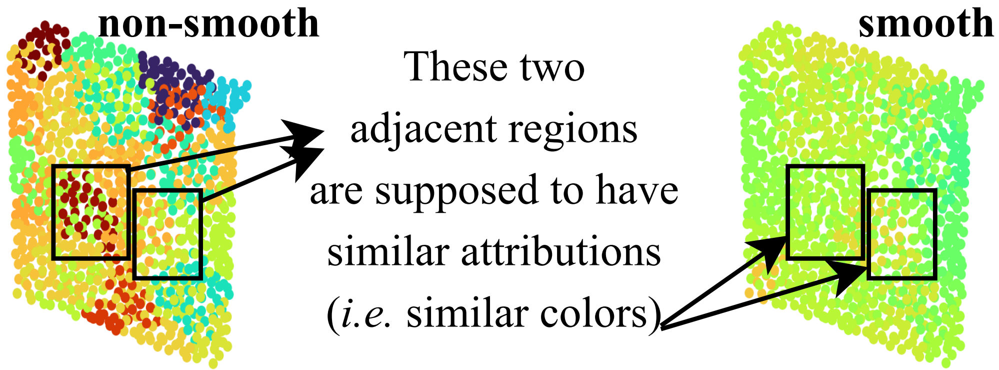
(a) Increasing **all regions'** linearity/planarity/scattering.

[1]Guinard and Landrieu. Weakly supervised segmentation-aided classification of urban scenes from 3d lidar point clouds. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII-1/W1:151–157, 2017.

[2]Demantké et al. Dimensionality Based Scale Selection in 3d LIDAR Point Clouds. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 3812:97–102, 2011.

Spatial smoothness

Most benchmark 3D datasets[1][2] only contain objects with **simple 3D structures**, except for special regions (*e.g.* edges), most adjacent regions of such simple objects usually have similar local 3D structures (*e.g.* surfaces).



(c) The smoothness of regional attributions.

Therefore, *most adjacent regions are supposed to have similar regional attributions, i.e. high smoothness.*

$$\textit{non-smoothness} = \mathbb{E}_{x\in X}\mathbb{E}_T\mathbb{E}_i\mathbb{E}_{j\in\boxed{\mathcal{N}(i)}}\left[\frac{|\phi_{x'}(i)-\phi_{x'}(j)|}{\boxed{Z_{\text{smooth}}}}\Big|_{x'=T(x)}\right]$$

a set of nearest point cloud regions of region i

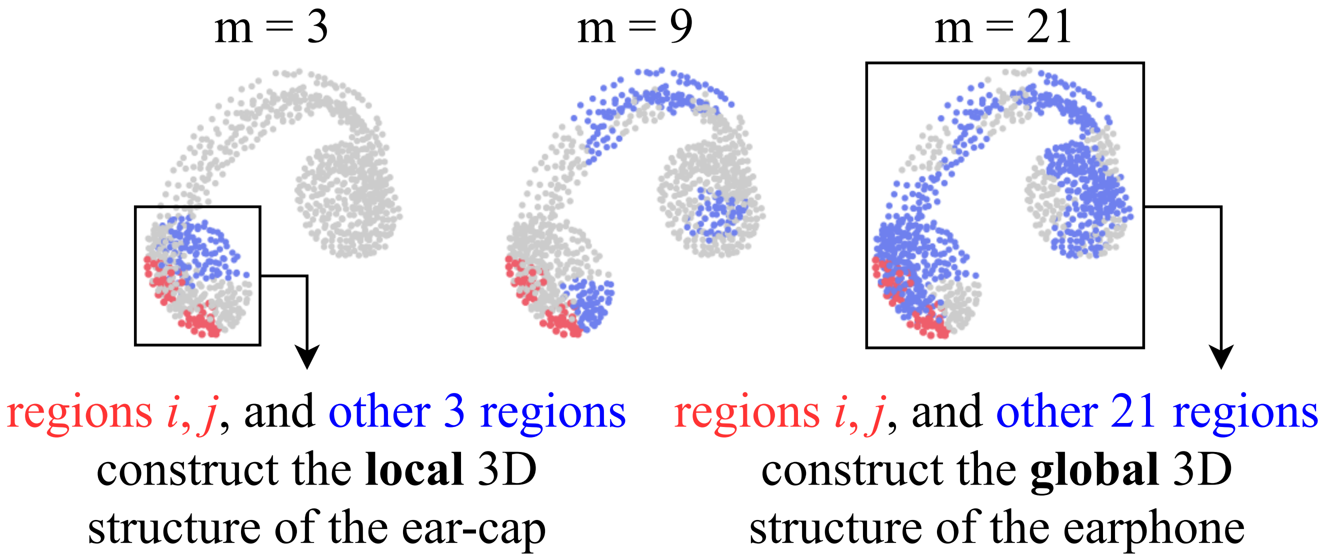
$$Z_{\text{smooth}} = \mathbb{E}_T[|v_{x'}(N)-v_{x'}(\emptyset)|_{x'=T(x)}]$$

is computed for normalization.

[1]Wu et al. 3d shapenets: A deep representation for volumetric shapes. In CVPR, 2015.
[2]Yi et al. A scalable active framework for region annotation in 3d shape collections. SIGGRAPH Asia, 2016.

Representation complexity

Input regions of a DNN do not work individually, but collaborate with each other to construct a specific 3D structure for inference.



(b) Visualizing the interaction of the *m*-th order.

m-th order interaction[1] between the regions i and j:

$$I^{(m)}(i,j) = \mathbb{E}_{S\subseteq N\setminus\{i,j\},|S|=m}\left[v(S\cup\{i,j\}) - v(S\cup\{i\}) - v(S\cup\{j\}) + v(S)\right]$$

Average strength of the m-th order interactions:

$$\boxed{I^{(m)} = \mathbb{E}_{x\in X}\left[\left|\mathbb{E}_{i,j}[I_x^{(m)}(i,j)]\right|\right]}$$

[1]Zhang et al. Interpreting and boosting dropout from a game-theoretic view. In ICLR, 2020.

Comparative studies

Explaining the regional sensitivity of DNNs

Dataset	Model	rotation sensitivity
ModelNet10	PointNet	0.159±0.070
	PointNet++	0.171±0.064
	PointConv	0.145±0.060
	DGCNN	0.174±0.075
	GCNN	0.174±0.067
	adv-GCNN[1]	0.034±0.012
ShapeNet part	PointNet	0.107±0.065
	PointNet++	0.142±0.057
	PointConv	0.168±0.073
	DGCNN	0.141±0.069
	GCNN	0.141±0.065
	adv-GCNN[1]	0.028±0.012

[1] adv-GCNN denoted the adversarially trained GCNN, which was supposed to be robust to rotation and translation.

Rotation robustness was the Achilles' heel of classic DNNs for 3D point cloud processing.

All DNNs were sensitive to rotations except for the adversarially trained GCNN.

Comparative studies

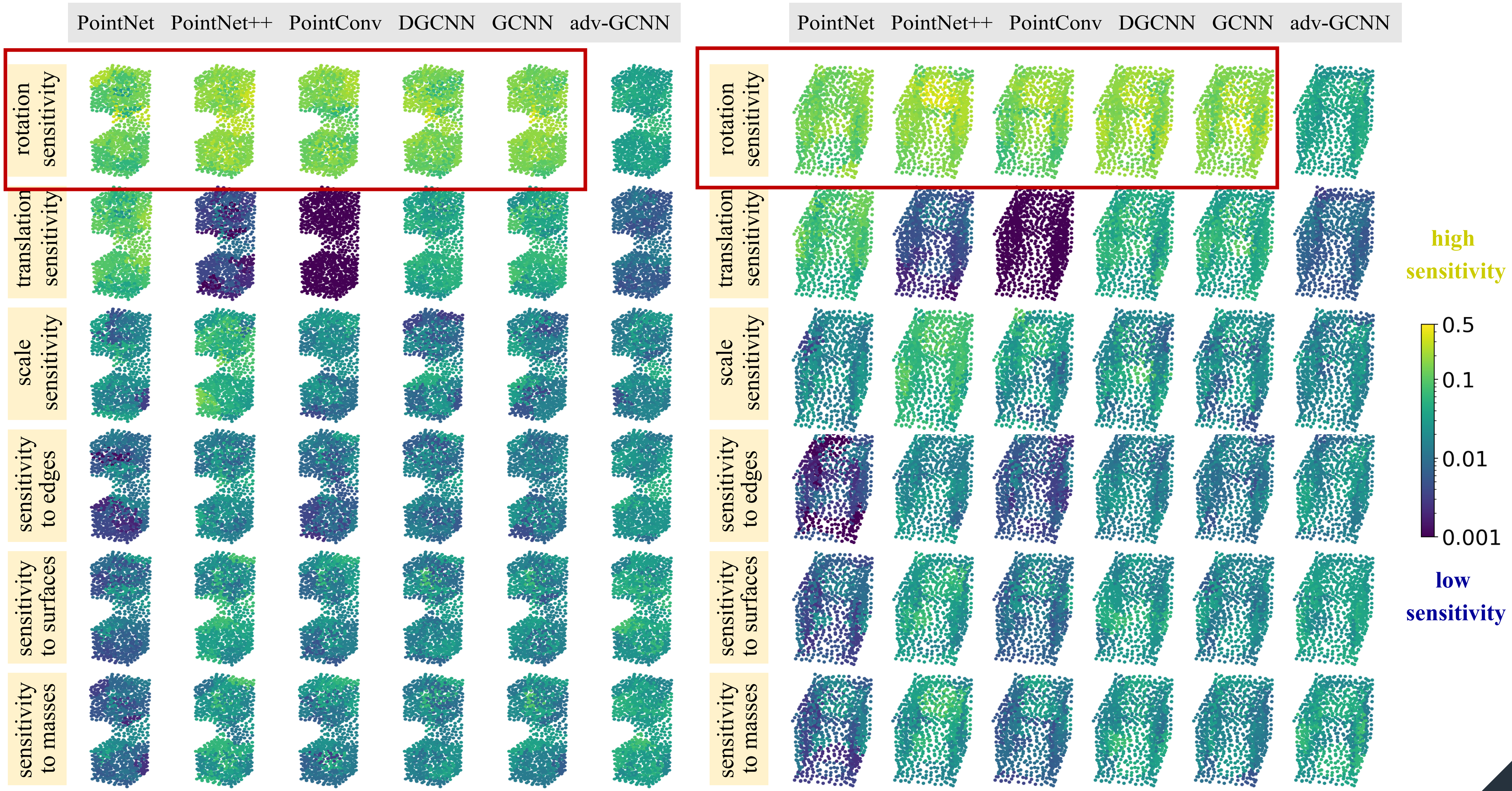
Explaining the regional sensitivity of DNNs

Table 1: Average sensitivities over all regions among all samples.

Dataset	Model	rotation sensitivity	translation sensitivity	scale sensitivity	sensitivity to edges	sensitivity to surfaces	sensitivity to masses
ModelNet10	PointNet	0.159±0.070	0.110±0.053	0.024±0.017	0.007±0.007	0.010±0.009	0.009±0.009
	PointNet++	0.171±0.064	0.004±0.004	0.054±0.027	0.018±0.011	0.026±0.016	0.029±0.019
	PointConv	0.145±0.060	2.3e-4±1.9e-4	0.027±0.019	0.010±0.007	0.015±0.011	0.017±0.013
	DGCNN	0.174±0.075	0.048±0.024	0.020±0.014	0.016±0.009	0.022±0.014	0.023±0.015
	GCNN	0.174±0.067	0.050±0.026	0.020±0.014	0.017±0.010	0.022±0.014	0.023±0.015
	adv-GCNN[1]	0.034±0.012	0.007±0.004	0.020±0.014	0.022±0.014	0.027±0.014	0.029±0.018
ShapeNet part	PointNet	0.107±0.065	0.071±0.032	0.023±0.020	0.005±0.005	0.004±0.004	0.005±0.005
	PointNet++	0.142±0.057	0.001±0.000	0.044±0.025	0.014±0.009	0.014±0.009	0.016±0.011
	PointConv	0.168±0.073	1.4e-5±2.5e-5	0.053±0.042	0.017±0.013	0.016±0.011	0.019±0.015
	DGCNN	0.141±0.069	0.067±0.033	0.020±0.015	0.014±0.011	0.013±0.011	0.016±0.013
	GCNN	0.141±0.065	0.072±0.038	0.021±0.015	0.014±0.011	0.013±0.010	0.016±0.015
	adv-GCNN[1]	0.028±0.012	0.009±0.008	0.025±0.020	0.028±0.022	0.024±0.015	0.028±0.019

[1] adv-GCNN denoted the adversarially trained GCNN, which was supposed to be robust to rotation and translation.

Comparative studies



Comparative studies

Explaining the regional sensitivity of DNNs

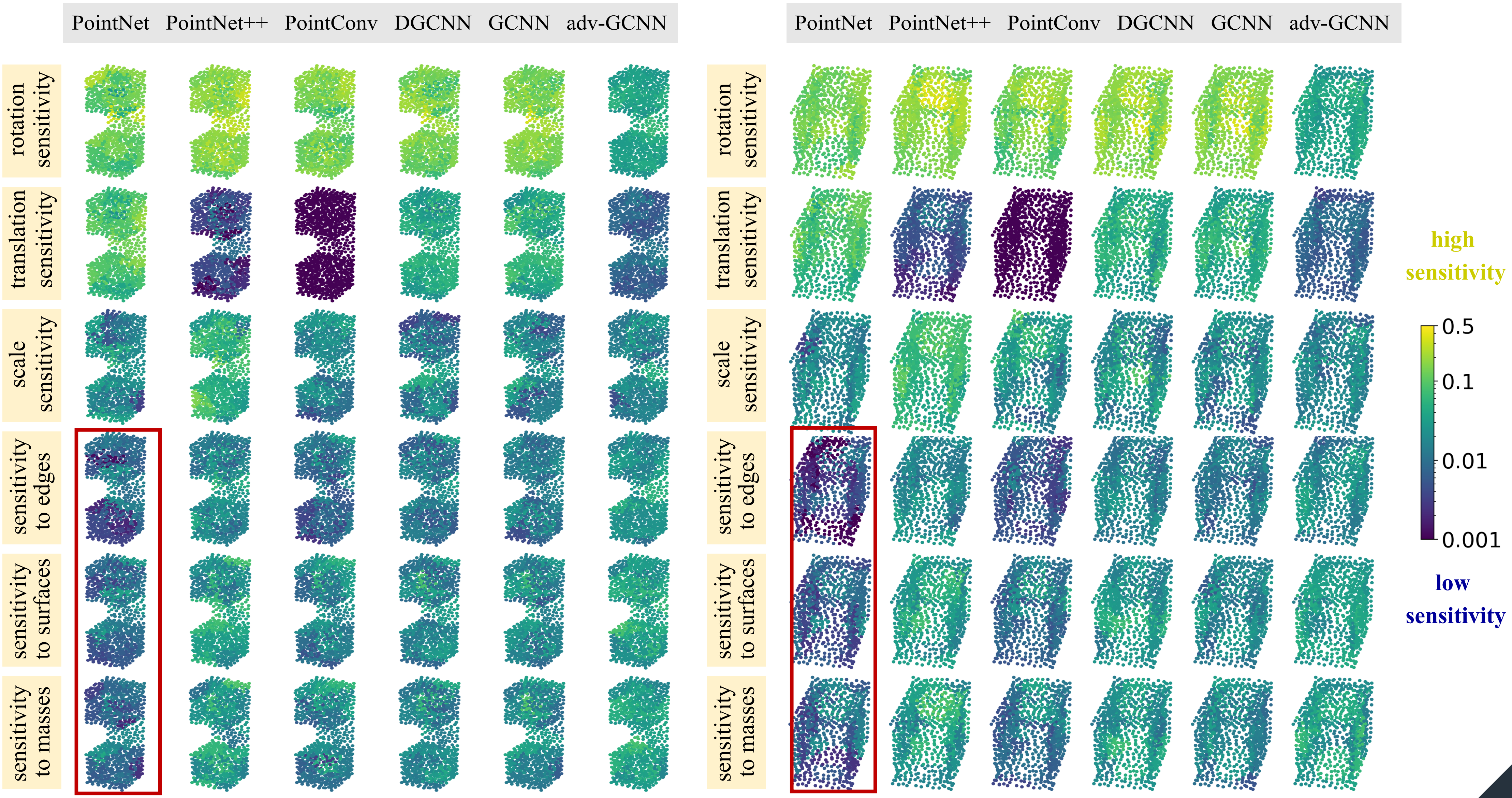
Dataset	Model	sensitivity to edges	sensitivity to surfaces	sensitivity to masses
ModelNet10	PointNet	0.007±0.007	0.010±0.009	0.009±0.009
	PointNet++	0.018±0.011	0.026±0.016	0.029±0.019
	PointConv	0.010±0.007	0.015±0.011	0.017±0.013
	DGCNN	0.016±0.009	0.022±0.014	0.023±0.015
	GCNN	0.017±0.010	0.022±0.014	0.023±0.015
	adv-GCNN[1]	0.022±0.014	0.027±0.014	0.029±0.018
ShapeNet part	PointNet	0.005±0.005	0.004±0.004	0.005±0.005
	PointNet++	0.014±0.009	0.014±0.009	0.016±0.011
	PointConv	0.017±0.013	0.016±0.011	0.019±0.015
	DGCNN	0.014±0.011	0.013±0.011	0.016±0.013
	GCNN	0.014±0.011	0.013±0.010	0.016±0.015
	adv-GCNN[1]	0.028±0.022	0.024±0.015	0.028±0.019

[1] adv-GCNN denoted the adversarially trained GCNN, which was supposed to be robust to rotation and translation.

PointNet failed to encode local 3D structures.

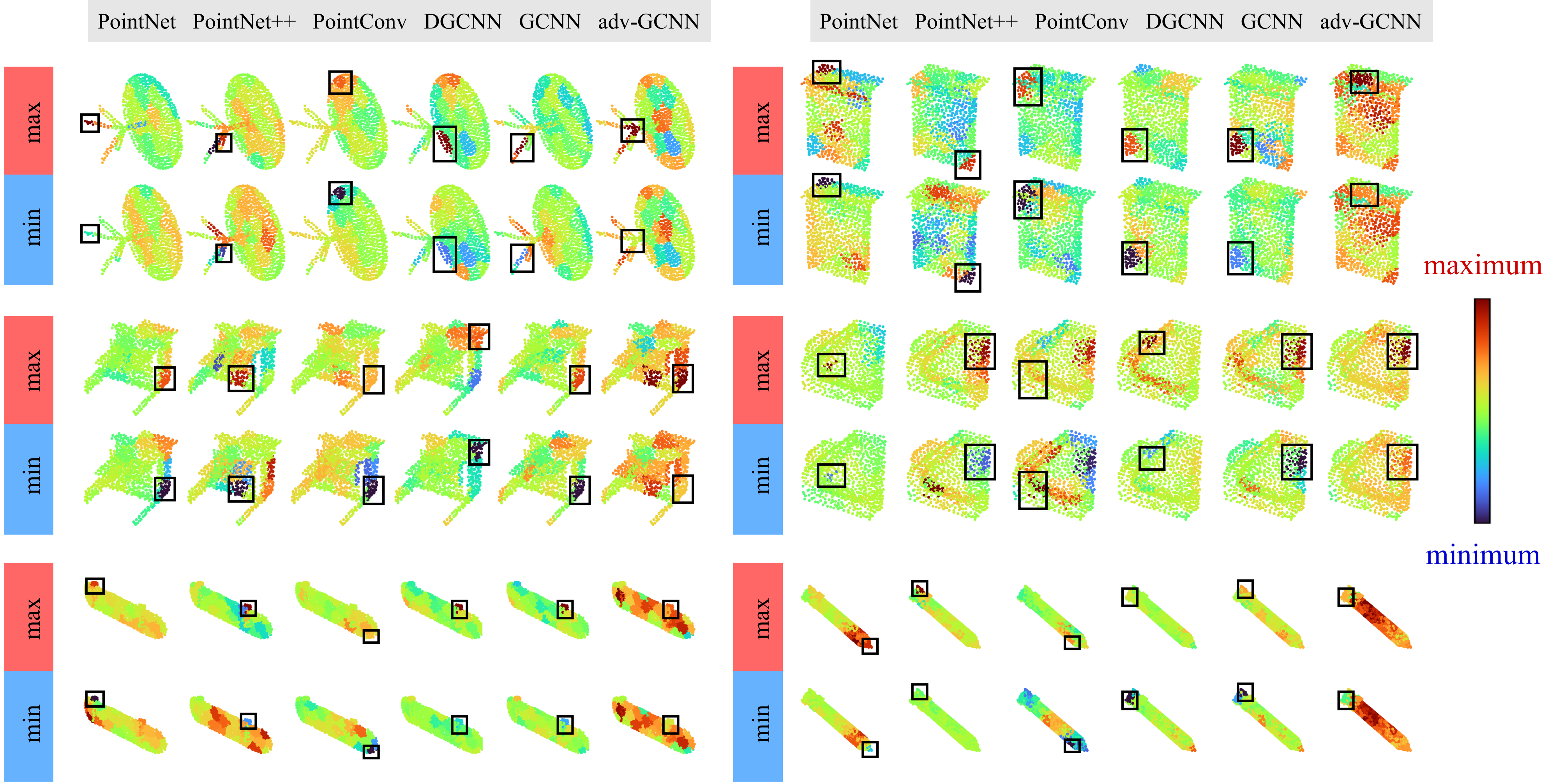
PointNet did not encode the information of neighboring points/regions.

Comparative studies



Comparative studies

Explaining the regional attributions



Most DNNs usually failed to extract rotation-robust features from 3D points at edges/corners.

Comparative studies

Pearson correlation coefficients between regional attributions and sensitivities.

Models	ModelNet10 dataset			ShapeNet part dataset		
	rotation sensitivity	translation sensitivity	scale sensitivity	rotation sensitivity	translation sensitivity	scale sensitivity
PointNet	$\boxed{0.648\pm0.266}$	0.637 ± 0.165	0.473 ± 0.194	$\boxed{0.528\pm0.278}$	0.549 ± 0.204	0.538 ± 0.275
PointNet++	$\boxed{0.811\pm0.123}$	0.415 ± 0.189	0.592 ± 0.142	$\boxed{0.629\pm0.154}$	0.266 ± 0.269	0.543 ± 0.171
PointConv	$\boxed{0.601\pm0.234}$	0.009 ± 0.179	0.473 ± 0.174	$\boxed{0.739\pm0.166}$	-0.006 ± 0.170	0.617 ± 0.168
DGCNN	$\boxed{0.788\pm0.111}$	0.622 ± 0.164	0.494 ± 0.224	$\boxed{0.725\pm0.176}$	0.649 ± 0.174	0.458 ± 0.201
GCNN	$\boxed{0.832\pm0.082}$	0.610 ± 0.131	0.464 ± 0.231	$\boxed{0.696\pm0.158}$	0.682 ± 0.198	0.431 ± 0.199
adv-GCNN[1]	$\boxed{0.488\pm0.167}$	0.298 ± 0.234	0.414 ± 0.256	$\boxed{0.343\pm0.234}$	0.255 ± 0.223	0.476 ± 0.304

[1] adv-GCNN denoted the adversarially trained GCNN.

- Most DNNs usually could not ignore features of rotation-sensitive points at edges and corners
- Adversarial training reduced the correlation between the regional sensitivity and the regional attribution

Comparative studies

Pearson correlation coefficients between regional attributions and sensitivities.

Models	ModelNet10 dataset			ShapeNet part dataset		
	rotation sensitivity	translation sensitivity	scale sensitivity	rotation sensitivity	translation sensitivity	scale sensitivity
PointNet	0.648±0.266	0.637±0.165	0.473±0.194	0.528±0.278	0.549±0.204	0.538±0.275
PointNet++	0.811±0.123	0.415±0.189	0.592±0.142	0.629±0.154	0.266±0.269	0.543±0.171
PointConv	0.601±0.234	0.009±0.179	0.473±0.174	0.739±0.166	-0.006±0.170	0.617±0.168
DGCNN	0.788±0.111	0.622±0.164	0.494±0.224	0.725±0.176	0.649±0.174	0.458±0.201
GCNN	0.832±0.082	0.610±0.131	0.464±0.231	0.696±0.158	0.682±0.198	0.431±0.199
adv-GCNN[1]	0.488±0.167	0.298±0.234	0.414±0.256	0.343±0.234	0.255±0.223	0.476±0.304

[1] adv-GCNN denoted the adversarially trained GCNN.

- Most DNNs usually could not ignore features of rotation-sensitive points at edges and corners
- Adversarial training reduced the correlation between the regional sensitivity and the regional attribution

Comparative studies

Explaining the spatial smoothness of DNNs

Table 3: The non-smoothness of attributions between adjacent regions.

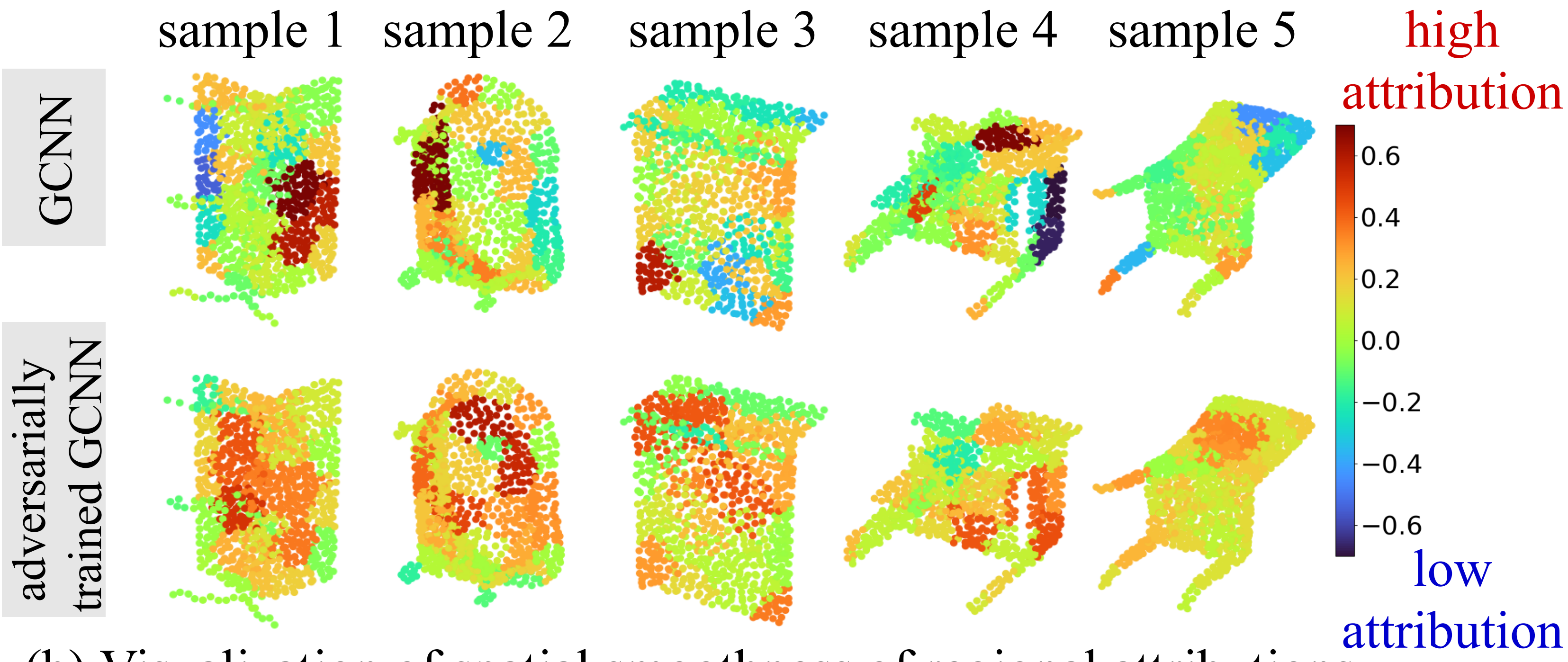
Models	ModelNet10 dataset		ShapeNet part dataset		ShapeNet part dataset (removing the biased category)	
	rotation	translation	rotation	translation	rotation	translation
PointNet	0.071±0.039	0.029±0.017	0.025±0.009	0.016±0.005	0.025±0.010	0.015±0.003
PointNet++	0.091±0.041	0.041±0.022	0.036±0.011	0.022±0.016	0.034±0.010	0.017±0.003
PointConv	0.047±0.014	0.056±0.108	0.080±0.019	0.040±0.017	0.081±0.020	0.039±0.018
DGCNN	0.071±0.024	0.031±0.010	0.047±0.019	0.026±0.017	0.044±0.017	0.021±0.005
GCNN	0.083±0.026	0.034±0.012	0.050±0.019	0.027±0.010	0.049±0.020	0.025±0.008
adv-GCNN[1]	0.029±0.012	0.030±0.013	0.054±0.110	0.056±0.114	0.022±0.008	0.023±0.008

[1] adv-GCNN denoted the adversarially trained GCNN.

Adversarial training increased the spatial smoothness of knowledge representations.

Comparative studies

Explaining the spatial smoothness of DNNs

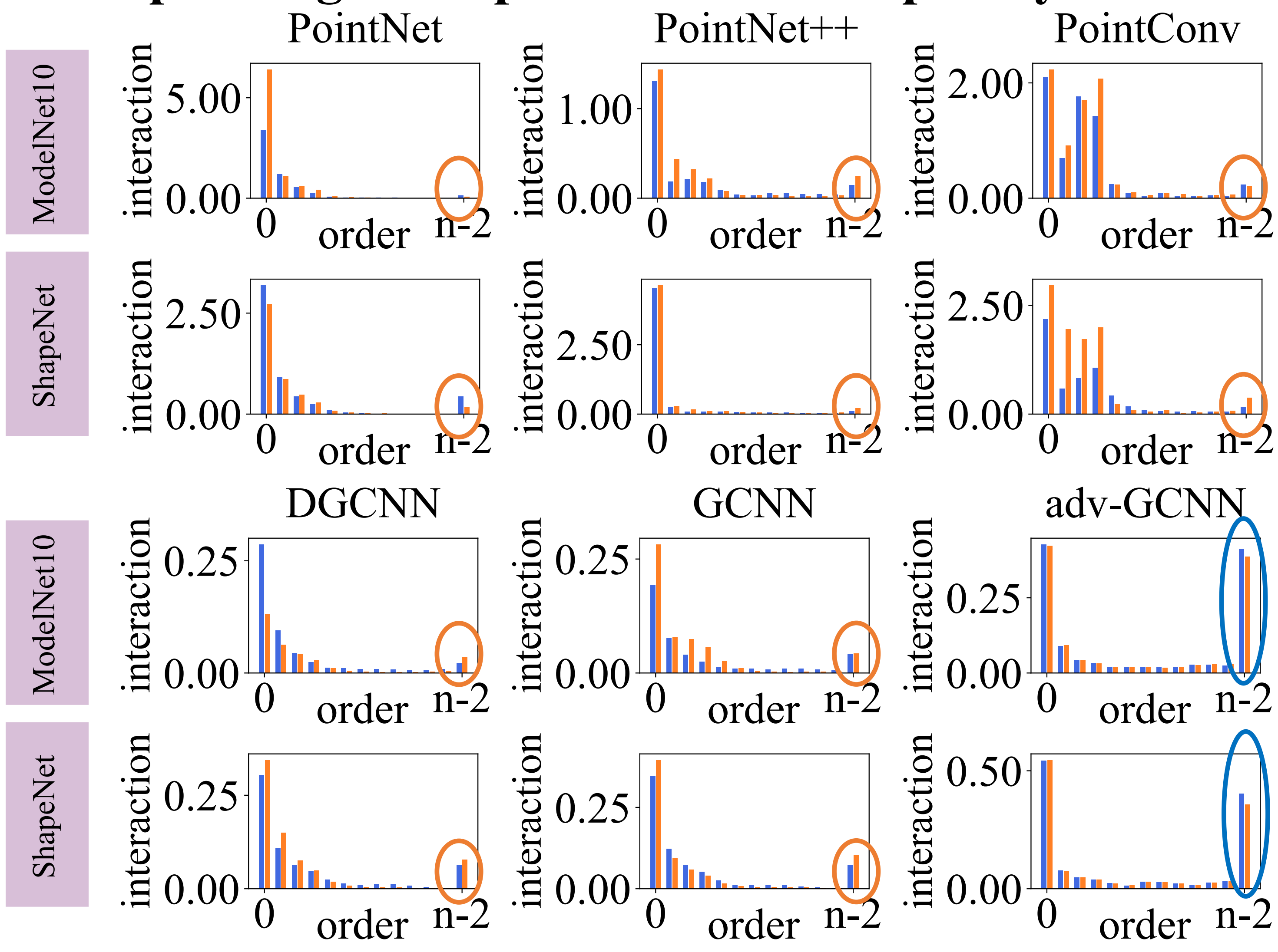


(b) Visualization of spatial smoothness of regional attributions.

Adversarial training increased the spatial smoothness of knowledge representations.

Comparative studies

Explaining the representation complexity of DNNs

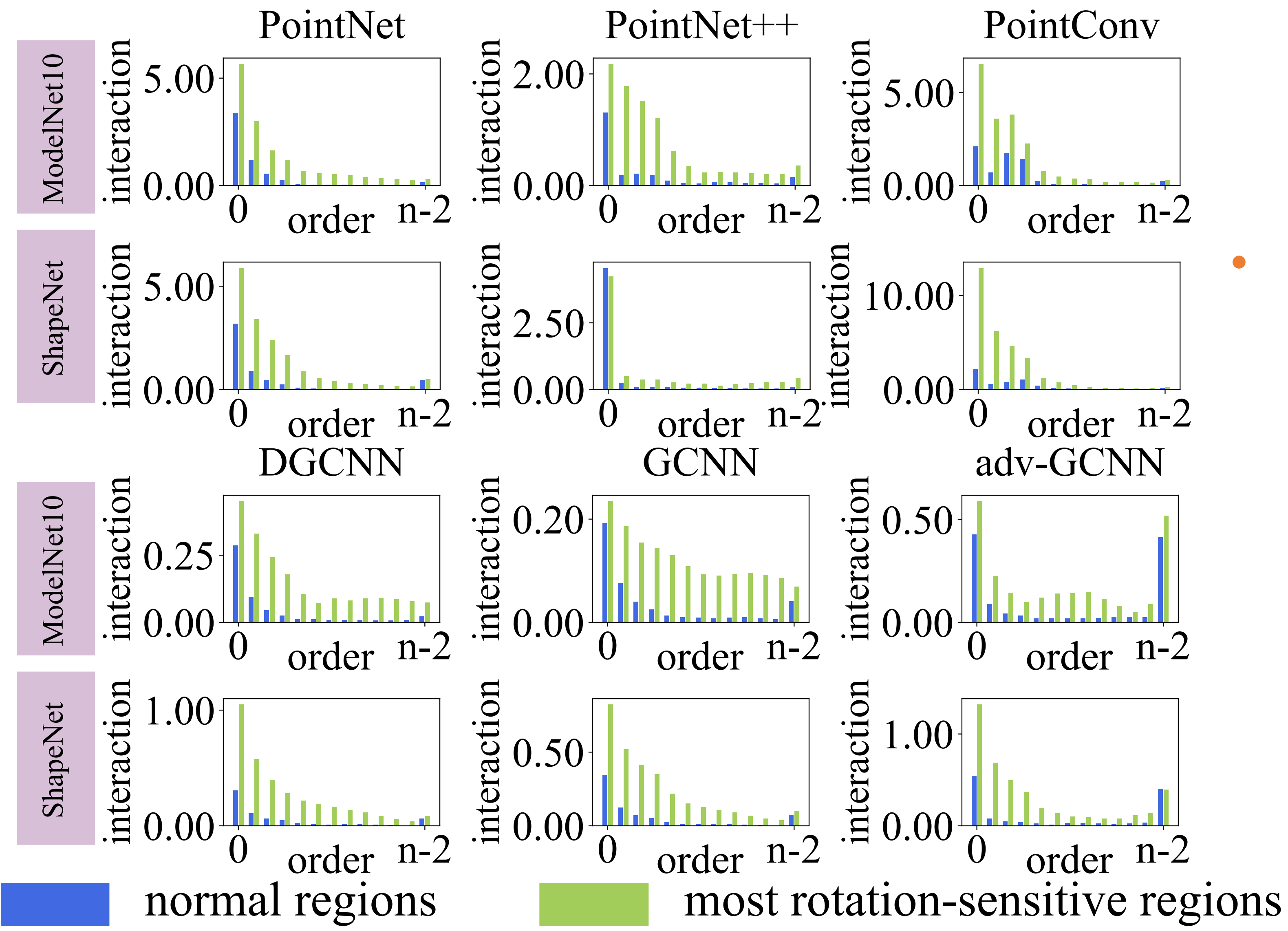


normal samples   adversarial samples (using rotations for attack, instead of perturbations)

- Most DNNs failed to encode high-order interactions
- Adversarial training increased the effects of extremely high-order interactions

Comparative studies

Explaining the representation complexity of DNNs



- Regions with out-of-distribution high-order interactions (*i.e.* abnormal complex and large-scale 3D structures) were more sensitive to the rotation than normal regions

THANK YOU !