Supplementary Materials of Improving Adversarial Robustness of 3D Point Cloud Classification Models

Guanlin Li[1,2], Guowen Xu[1,*], Han Qiu[3], Ruan He[4], Jiwei Li[5,6], and Tianwei Zhang[1]

[1] Nanyang Technological University, [2] S-Lab, NTU,
[3] Tsinghua University, [4] Tencent, [5] Shannon.AI, [6] Zhejiang University,
{guanlin001,guowen.xu, tianwei.zhang}@ntu.edu.sg,
qiuhan@tsinghua.edu.cn, ruanhe@tencent.com,
jiwei_li@shannonai.com
[*] Corresponding author

1 Mutual Information Maximization

Theorem 1 *Let f be a function that maps a point cloud to the feature space, and Q be the distribution of clean point clouds. S is sampled from Q. $Q^k(S,\epsilon)$ is the distribution of noisy point clouds, in which each element S_k is perturbed from S with an additional noise ϵ, and the difference of numbers of points between S and S_k is smaller than a constant k,* i.e., *$-k \leq |S_k| - |S| \leq k$. Then for every $S \sim Q$ and $S_k \sim Q^k(S,\epsilon)$, the mutual information $I(S_k, f(S))$ has a lower bound, which is negatively correlated with the k-measurement $M_k(f, S, S_k)$.*

Proof. According to the definition of mutual information, we have the following equation:

$$I(S_k, f(S)) = H(f(S)) - H(f(S)|S_k) = \\ -\sum_{1}^{|Q|} \frac{1}{|Q|} \log f(S) - H(f(S)|S_k).$$

We use $B(S,\epsilon)$ to denote a hyper-sphere whose center is S and radius is $\|\epsilon\|$. So the second term of the above expression can be rewritten as follows:

$$-H(f(S)|S_k) = \sum_k \sum_{S_k \sim Q^k(S,\epsilon)} Pr[f(S), S_k] \log Pr[f(S)|S_k] \\ \geq \sum_k \int_\epsilon \int_{S_k \sim B(S,\epsilon)} \frac{1}{M_k(f,S,S_k)} \log \frac{|Q|}{M_k(f,S,S_k)}.$$

The mutual information can be further derived as follows:

$$I(S_k, f(S)) \geq -\sum_{1}^{|Q|} \frac{1}{|Q|} \log f(S) + \\ \sum_k \int_\epsilon \int_{S_k \sim B(S,\epsilon)} \frac{1}{M_k(f,S,S_k)} \log \frac{|Q|}{M_k(f,S,S_k)}.$$

This means the lower bound of $I(S_k, f(S))$ is increased when $M_k(f, S, S_k)$ is smaller.

Works & Venue	Type	Models	Datasets	Attacks	Attack Type
ICCV'19 [13]	Defense	PointNet, PointNet++, DGCNN	ModelNet40	AIC, AIH, APP, SMA	P, A, D
ICCV'19 [12]	Attack	PointNet, PointNet++, DGCNN	3DMNIST, ModelNet40	SMA	D
ICIP'19 [3]	Defense & Attack	PointNet, PointNet++	ModelNet40	FGSM, I-FGSM, JSMA	P
CVPR'19 [9]	Attack	PointNet, PointNet++, DGCNN	ModelNet40	AIC, AIH, APP	P, A
MM'20 [5]	Attack	PointNet, PointNet++, DGCNN	ModelNet40	FGSM, PGD, CW	P
AAAI'20 [7]	Attack	PointNet++	ModelNet40	kNN	P
CVPR'20 [1]	Defense	PointNet, PointNet++	ModelNet40	FGSM, I-FGSM, PGD, MI-PGD	P
ECCV'20 [2]	Attack	PointNet, PointNet++, DGCNN	ModelNet40	AdvPC	P
ICCV'21 [4]	Defense	PointNet	ModelNet40	FGSM, PGD	P
TPAMI [8]	Attack	PointNet, PointNet++, DGCNN	ModelNet40	$GeoA^3$	P
ArXiv [11]	Defense & Attack	PointNet	ModelNet40	PG, PD, PA	P, A, D
ArXiv [6]	Defense & Attack	PointNet++, GvG-P, DUP-Net	ModelNet40	FGSM, BIM, MIM	P
ArXiv [10]	Defense	PointNet, PointNet++, DGCNN, RS-CNN	ModelNet10, ModelNet40	APP, kNN, SMA	P, D, B
Ours	Defense	PointNet, PointNet++, DGCNN, DUP-Net	ModelNet10, ModelNet40	AIC, AIH, APP, SMA, BIM, AdvPC	P, A, D, B

Table 1: Summary of evaluations in prior works and this paper. For attack type: P - point perturbing; A - point adding; D - point dropping; B - blackbox attack

2 Comparison With Previous Works

Table 1 summarizes the models, datasets and attacks adopted in our experiments, as well as comparisons with prior works. We claim that our evaluations are more comprehensive than existing studies about point cloud robustness.

3 PointNet++ under attacks

Network Structure	Training Strategy	Clean	Attacks					
			SMA−40	APP	AIC	AIH	**AAUA**	**LAUA**
PointNet++ (MSG)	Normal	89.77	62.18	0.00	0.00	0.00	15.55	0.00
	PointCutMix-R	92.57	85.92	0.00	0.16	2.15	22.06	0.00
	AT-BIM	88.47	64.00	0.00	0.00	0.00	16.00	0.00
PointNet++ (SSG)	Normal	89.85	56.45	0.00	0.00	0.00	14.11	0.00
	PointCutMix-R	92.78	86.57	0.00	1.83	2.88	22.82	0.00
	AT-BIM	89.53	71.51	0.00	0.00	0.00	17.88	0.00

Table 1: Accuracy of PointNet++ under untargeted attacks (%). All results are running results.

We further compare the accuracy of two types of PointNet++ under attacks. The results are shown in Table 1. For both types of PointNet++, training models with mix-up samples can significantly improve the accuracy under the dropping point attack, as mix-up samples can be seen as clean points dropped a lot of original points. However, PointNet++ cannot defend against adding perturbation attacks and adding additional points attacks. As we analyzed before, PointNet++ uses each point and its neighbors sampled based on distances coordinates to generate local features directly. When sampling neighbors on perturbed point clouds or point clouds with additional points, PointNet++

will use more noisy points to generate local features causing noise accumulating. Comparing with previous works, we find when using targeted attacks to attack PointNet++, the accuracy under attacks are significantly higher than results in Table 1. It is easy to understand that untargeted attacks are more powerful, and PointNet++ does not always predict adversarial examples as labels the adversary wants. For an adversary who wants the model to give wrong labels instead of specific labels, attacking PointNet++ is uncomplicated. Since the structure of PointNet++ is fragile under attacks, we do not apply our AMS on it.

4 Experiments of AMS

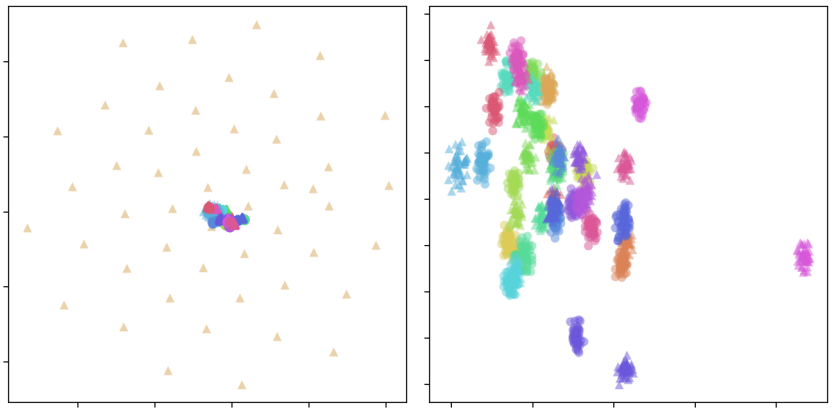
When comparing normally trained models and models trained with AT-BIM, we find that DGCNN achieves the highest clean accuracy. However, the DGCNN does not outperform our CCN under all four white-box attacks. On the other hand, the PointNet shows the worst performance. When comparing models trained with AT-BIM and our AMS, we can clearly notice that the AMS generalizes well to other model structures. Our CCN outperforms other baselines on clean accuracy and many white-box attacks. It achieves not only the highest **AAUA** but also the highest **LAUA**. It means that our CCN can work with the AMS together in harmony. In summary, for each architecture, AMS gives the best performance compared to Normal or AT-BIM training. To sum up, the integration of CCN and AMS is the most robust solution.

Network Structure	Training Strategy	Clean Sample	Adversarial Examples					
			SMA−40	APP	AIC	AIH	**AAUA**	**LAUA**
PointNet	Normal	88.76	41.88	55.64	49.68	43.43	47.66	41.88
	AT-BIM	88.23	45.41	85.39	84.98	86.36	75.54	45.41
	AMS	89.45	48.99	87.01	**86.49**	**87.26**	77.44	48.99
DGCNN	Normal	91.03	65.87	46.10	54.06	48.78	53.70	46.10
	AT-BIM	91.27	66.68	89.98	81.37	76.99	78.76	66.68
	AMS	**92.21**	75.41	**90.83**	85.47	83.93	83.91	75.41
CCN	Normal	90.87	67.94	57.47	61.04	53.37	59.96	53.37
	AT-BIM	90.05	67.37	88.80	83.77	79.75	79.92	67.37
	AMS	**92.41**	**77.72**	**90.50**	**86.09**	84.05	**84.74**	**77.72**

Table 2: Model accuracy for different solutions under the white-box attacks (%).

5 t-SNE Results Zoom Out

We plot all 40 classes (represented with different colors), and each class contains 50 point clouds from ModelNet40. Circles and triangles denote the clean and perturbed point clouds, respectively. From the Fig. 1, the DGCNN is not as robust as CCN. When we add perturbation to point clouds from a class, the features scatter in the feature space. However, our CCN will not be influenced by the perturbation significantly, which indicates that our CCMs in CCN can efficiently decrease the noise in the inputs.

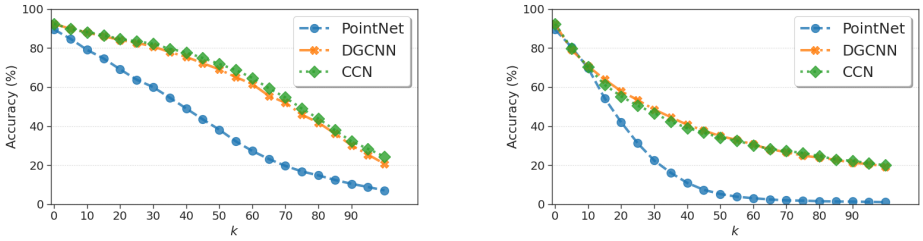


(a) DGCNN (b) CCN

Fig. 1: Feature map visualization.

6 Robustness under Different Attack Budgets

Furthermore, we show the accuracy of models trained with our AMS under SMA−k and BIM−k with different k in Fig. 2. When models are attacked by SMA−k, the PointNet is more fragile than other two models, resulting it has the lowest accuracy. The CCN outperforms the DGCNN with the k increasing becoming more clearly. When we attack models with BIM−k, we find that when the k is small, the accuracy of three models are very close. With the k increasing, the accuracy of the PointNet drops very quickly. As for CCN and DGCNN, the accuracy of DGCNN is higher than the accuracy of CCN at the start. However, when the k is higher than 60, the CCN starts to outperform the DGCNN. Both of them achieve higher accuracy than the PointNet. Overall, our CCN is the most robust one.



(a) Accuracy under SMA attack with different K (b) Accuracy under BIM attack with different K

Fig. 2: Accuracy of models under SMA−K and BIM−K. All models are trained with our AMS.

7 Feature Distances under Different Perturbation Budgets

In Figs. 3 to 7, we compare feature distances under different perturbation strength. We adjust perturbation based on its variance. For each trail, we run 10 times and calculate the average distance. The results indicate that the scale of perturbation only has trivial influence of the feature distances. Our CCN can reduce the distances with the layer going deeper. Models trained with AMS can obtain smaller distance under all cases. Combining the above two phenomena, we can claim that our CCN and AMS can reduce feature distances and achieve higher mutual information. So both of them can improve model's robustness and be harmonious with our theoretical analysis.

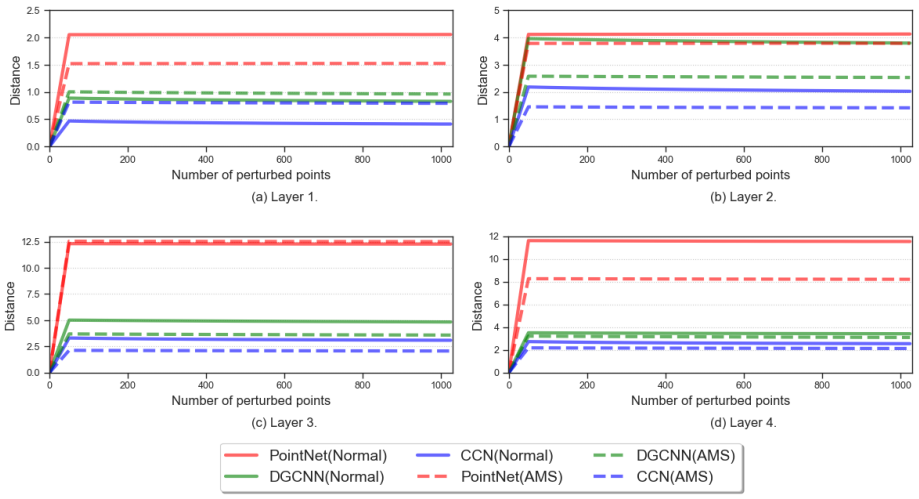


Fig. 3: Cosine distance of features between clean and perturbed samples from different layers. The perturbation is generated from a Gaussian distribution with $\mu = 0$ and $\sigma = 0.01$.

8 Model Size

Table 3 compares the model size and number of parameters for different point cloud models. We observe that CCN is slightly bigger than DGCNN due to the introduction of the CCM. Its size is still smaller than PointNet++. Nevertheless, CCN gives the best robustness among these models.

9 Comparing AT with multiple types of attacks.

We consider a stronger baseline method, "Multiple Types of Attacks" (MTA), where the robust model is trained with two types of AEs (BIM-20 attack and SMA-20 attack) together. Table 4 compares AMS with this MTA strategy. We have the following observations. (1) For the clean accuracy, CCN is better than MTA, as it uses the mix-up

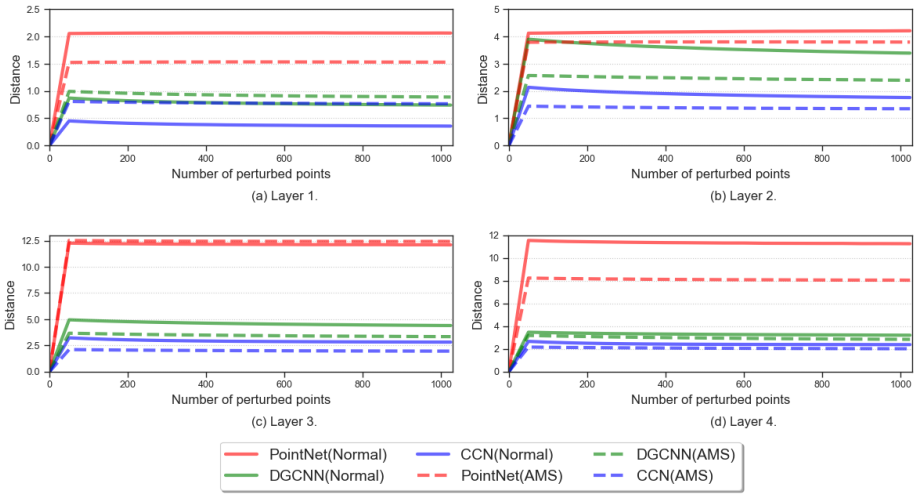


Fig. 4: Cosine distance of features between clean and perturbed samples from different layers. The perturbation is generated from a Gaussian distribution with $\mu = 0$ and $\sigma = 0.03$.

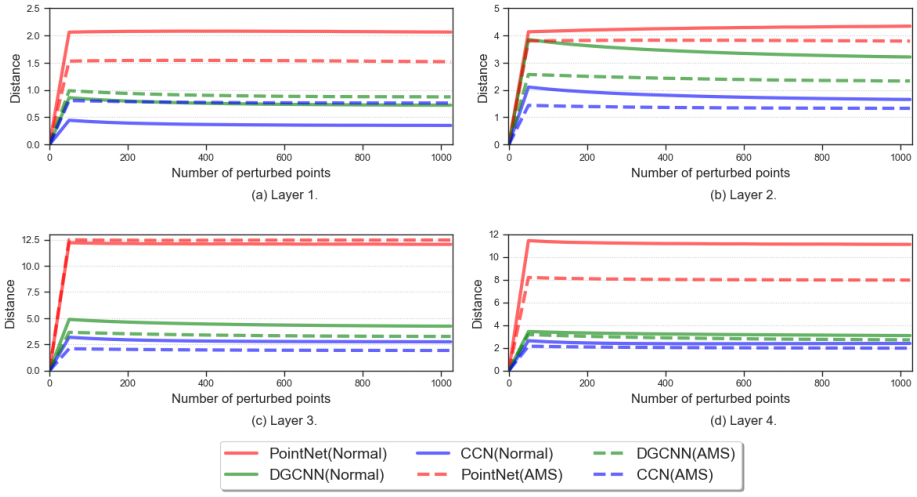


Fig. 5: Cosine distance of features between clean and perturbed samples from different layers. The perturbation is generated from a Gaussian distribution with $\mu = 0$ and $\sigma = 0.05$.

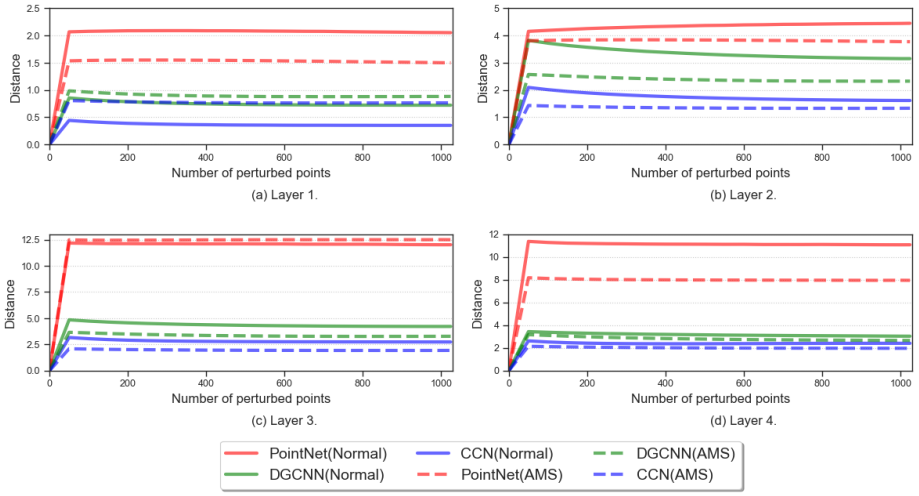


Fig. 6: Cosine distance of features between clean and perturbed samples from different layers. The perturbation is generated from a Gaussian distribution with $\mu = 0$ and $\sigma = 0.07$.

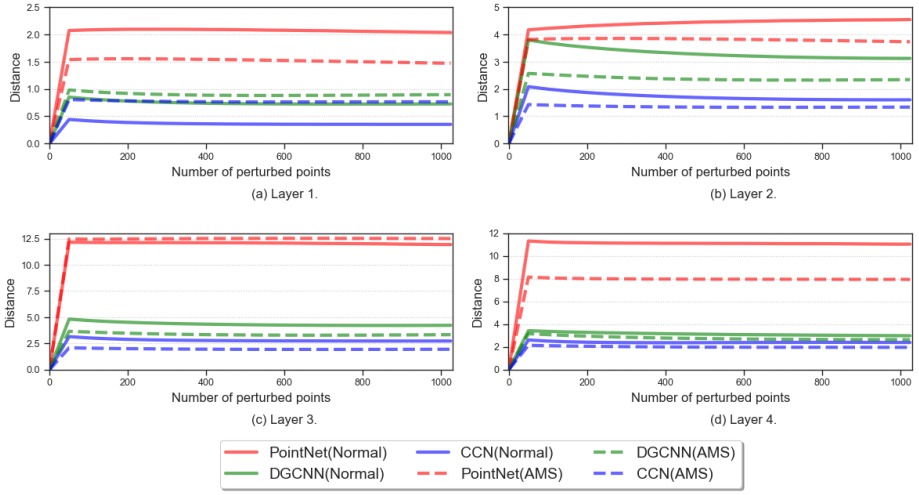


Fig. 7: Cosine distance of features between clean and perturbed samples from different layers. The perturbation is generated from a Gaussian distribution with $\mu = 0$ and $\sigma = 0.1$.

Model	Model size (MB)	# of Para (million).
PointNet	9.4	0.80
PointNet++	12.0	1.02
DGCNN	11.0	0.94
CCN	11.6	0.98

Table 3: The numbers of model parameters and model sizes for different point cloud models.

samples in the training process. (2) For the robustness, MTA only performs better than CCN for the SMA-20 attack, since it adopts this attack for adversarial training, and the robustness is overfitted on these samples, while AMS uses SMA-10. For other attacks, AMS outperforms MTA. AMS also gives the best **AAUA** and **LAUA**. This indicates AMS is still the better training strategy.

Model	Strategy	Clean Sample	Adversarial Examples					
			SMA-20	APP	AIC	AIH	**AAUA**	**LAUA**
PointNet	MTA	87.70	76.54	86.97	85.63	86.44	83.90	76.54
	AMS	89.45	69.03	87.01	**86.49**	**87.26**	82.46	69.03
DGCNN	MTA	90.79	**85.71**	87.95	82.35	76.95	83.24	76.95
	AMS	**92.21**	84.09	**90.83**	85.47	83.93	**88.58**	**83.93**

Table 4: Comparison of more baselines.

10 Results on ModelNet10

Defense Solutions	Clean Sample	Adversarial Examples					
		SMA-40	APP	AIC	AIH	**AAUA**	**LAUA**
PointNet + AMS	84.82	45.98	75.22	**73.33**	**69.53**	66.02	45.98
DGCNN + AMS	**93.64**	**81.36**	75.89	63.84	57.25	69.59	57.25
CCN + AMS	92.75	72.10	**78.01**	70.65	61.72	**70.62**	**61.72**

Table 5: Results on ModelNet10.

We verify the effectiveness of our CCN and AMS on ModelNet10 in Table 5. As ModelNet10 can be seen as a toy dataset, we do not fine-tune our training hyperparameters and only verify our proposed methods. The results verify that our methods can still work on other dataset. Because ModelNet10 is a small dataset, models heavily overfit the AEs and the robustness is not as high as on the ModelNet40.

References

1. Dong, X., Chen, D., Zhou, H., Hua, G., Zhang, W., Yu, N.: Self-robust 3d point recognition via gather-vector guidance. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11513–11521. IEEE (2020)

2. Hamdi, A., Rojas, S., Thabet, A.K., Ghanem, B.: Advpc: Transferable adversarial perturbations on 3d point clouds. In: Proc. of the ECCV. vol. 12357, pp. 241–257 (2020)
3. Liu, D., Yu, R., Su, H.: Extending Adversarial Attacks and Defenses to Deep 3D Point Cloud Classifiers. In: Proc. of the ICIP (2019)
4. Lorenz, T., Ruoss, A., Balunović, M., Singh, G., Vechev, M.: Robustness certification for point cloud models. arXiv preprint arXiv:2103.16652 (2021)
5. Ma, C., Meng, W., Wu, B., Xu, S., Zhang, X.: Efficient Joint Gradient Based Attack Against SOR Defense for 3D Point Cloud Classification. In: Proc. of the MM. pp. 1819–1827 (2020)
6. Sun, J., Koenig, K., Cao, Y., Chen, Q.A., Mao, Z.M.: On adversarial robustness of 3d point cloud classification under adaptive attacks. arXiv preprint arXiv:2011.11922 (2020)
7. Tsai, T., Yang, K., Ho, T.Y., Jin, Y.: Robust adversarial objects against deep learning models. In: Proc. of the AAAI. pp. 954–962 (2020)
8. Wen, Y., Lin, J., Chen, K., Jia, K.: Geometry-aware Generation of Adversarial and Cooperative Point Clouds. CoRR **abs/1912.11171** (2019)
9. Xiang, C., Qi, C.R., Li, B.: Generating 3D Adversarial Point Clouds. In: Proc. of the CVPR (2019)
10. Zhang, J., Chen, L., Ouyang, B., Liu, B., Zhu, J., Chen, Y., Meng, Y., Wu, D.: PointCutMix: Regularization Strategy for Point Cloud Classification. CoRR **abs/2101.01461** (2021)
11. Zhang, Q., Yang, J., Fang, R., Ni, B., Liu, J., Tian, Q.: Adversarial attack and defense on point sets. CoRR **abs/1902.10899** (2019)
12. Zheng, T., Chen, C., Yuan, J., Li, B., Ren, K.: PointCloud Saliency Maps. In: Proc. of the ICCV (2019)
13. Zhou, H., Chen, K., Zhang, W., Fang, H., Zhou, W., Yu, N.: DUP-Net: Denoiser and Upsampler Network for 3D Adversarial Point Clouds Defense. In: Proc. of the ICCV (2019)