# Denoise and Contrast for Category Agnostic Shape Completion Supplementary Material

Antonio Alliegro[1] Diego Valsesia[1] Giulia Fracastoro[1]
Enrico Magli[1] Tatiana Tommasi[1,2]

[1]Politecnico di Torino, Italy [2]Italian Institute of Technology

{name.surname}@polito.it

## 1. Additional Benchmark Analysis

In the main paper we mentioned a recent work on point-cloud shape completion based on the idea of separated feature aggregation [7]. It uses local features to represent the known part and keep the original details, while global features are exploited for the missing part to describe the latent underlying surface. Since the proposed network is designed to reconstruct the complete shape with ground truth clouds containing 16384 points, we operated some minimal changing on the architecture to get a fair comparison on 2048-points ground truth without corrupting its original nature. Specifically, we started from the Residual Feature Aggregation (RFA) method, in which the missing part is represented with residual features between the global shape and the known part. We considered two variants: in the first one we generated a coarse output of 1024 points, then refined to 2048 points by the folding module inherited from PCN [6]. In the second one we dropped the folding module and we selected the top scored 2048 points at the final attention module as prediction. We also experimentally verified that the repulsion loss of the method becomes detrimental when dealing with a low-resolution ground truth, so we did not include it in the learning process. This second variant obtained better results than the first and the corresponding CD are collected in Table 1. The comparison indicates that both PF-Net and DeCo largely outperform RFA. The renderings in Figure 1 confirm that RFA produces a reasonable overall object shape, but the missing part is often noisy and reconstructed with artifacts.

| Category | RFA [7] | PF-Net [2] | DeCo |
|---|---|---|---|
| Airplane | 26.747 | 10.805 | **10.003** |
| Bag | 40.153 | 38.485 | **28.508** |
| Cap | 47.150 | 50.450 | **36.436** |
| Car | 59.167 | **21.640** | 22.963 |
| Chair | 29.227 | 19.490 | **16.428** |
| Lamp | 64.243 | 42.910 | **24.150** |
| Laptop | 27.880 | **11.220** | 12.706 |
| Motorbike | 71.623 | 19.905 | **19.136** |
| Mug | 80.200 | **31.880** | 34.239 |
| Pistol | 23.783 | **10.885** | 12.266 |
| Skateboard | 127.413 | 12.365 | **9.861** |
| Table | 31.903 | 20.845 | **17.120** |
| Guitar | 13.357 | **4.425** | 4.482 |
| Overall | 36.773 | 20.445 | **16.517** |

Table 1. *Known Categories - Quantitative*. Chamfer Distance on the point cloud missing region scaled by $10^4$. The lower, the better.

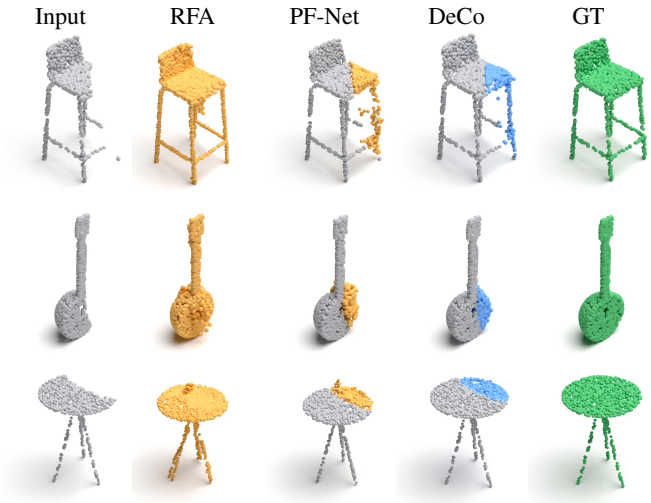


Figure 1. *Known Categories - Qualitative*. In order from top to bottom: chair, guitar, table. RFA shows artifacts and less precise reconstruction than PF-Net and DeCo.

## 2. Decoder Output & Frame Dimension

Our decoder includes two SAG Pool layers [3], whose purpose is to reduce the number of input points down to the number of points of the missing part. We exploit a hierarchical pooling logic in order to predict at different decoder depth both the *frame + missing region* $\mathbf{Y}_{fm}$, and the *miss-*

*ing region* $\mathbf{Y}_m$. The total number of output points at the two prediction heads depends on the number of pooled feature-space nodes, which are then decoded from the feature to the 3D space. While the choice of the $N_2$ parameter is constrained by the *missing part* ground truth size ($N_2 = M$), we are free to tune $N_1$, as long as it holds $N_1 \geq M + F$. As specified in the main paper, in case of the standard single hole analysis, we set $M = F = 512$, so $\boldsymbol{X}_m$ has dimension $(512, 3)$, while $\boldsymbol{X}_{fm}$ has dimension $(1024, 3)$. For the decoder we had $N_1 = 1280$ and $N_2 = 512$, thus resulting in $\mathbf{Y}_{fm}$ and $\mathbf{Y}_m$ respectively with size $(1280, 3)$ and $(512, 3)$.

In Table 2 we show the results obtained by varying $\{N_1, F\}$: the CD are always lower than those of the best competitor PF-Net (20.445). Moreover the obtained results confirm the effectiveness our parameter choice.

In the main paper we also discussed two robustness tests. In the case of a *single large hole* ($50\%$ of point cloud missing, 1024 points out of a whole shape of 2048 points) we simply dropped the frame condition and removed the two SAG Pool layers from the decoder, thus we did not use the frame auxiliary prediction in training. Despite this simplification, DeCo consistently outperforms its best competitor PFNet, demonstrating the effectiveness of our architecture and training procedure also when half of the complete shape is missing. In the case of *two holes*, each covering $12.5\%$ of the point cloud, we kept the condition $M = F$, so each crop consists of 256 points with their respective frame of equal cardinality out of a whole shape of 2048 points. The results in the main paper have shown how recovering the complete shape from a multiple-drilled partial input is way harder than recovering from a single-drilled shape, nevertheless DeCo is still able to outperform all the considered baselines.

## 3. Contrastive Learning: Quadruplets vs Pairs

In the main paper we described our strategy to extract global information from the point clouds via contrastive learning. Specifically we adopted a variant based on sample quadruplets, rather than on pairs as in the standard contrastive learning solution [1]. We present here a detailed analysis of this choice. More precisely, Table 3 shows how using sample pairs can still provide good results, but passing from pairs to quadruplets allows us to work with a more manageable batch size, while also providing a further improvement in the reconstruction accuracy.

## 4. DGCNN for Denoising

In all DeCo experiments in the main paper we used at the local encoder the powerful Graph-Convolutional Point Denoising network (GPDNet) proposed in [4]. Here we also present the completion results obtained by replacing it with a more conventional DGCNN [5] encoder. All the other components of DeCo remain the ones already described, and we follow the same pre-training procedure adopted in the main paper for the denoising task (Gaussian noise, mean=0, standard deviation=0.02) of the simplified local encoder. The results in Table 4 show that the obtained DGCNN-based lighter version of DeCo still provides state-of-the-art performance, highlighting the effectiveness of our training strategy regardless of the specific adopted graph convolution blocks and backbone. As reference we also report the PF-Net baseline results and the number of parameters for all the considered variants which confirms the significant advantage of DeCo with respect to its best competitor also in terms of parameter cardinality.

| $\mathbf{N_1}$ | **M=512** | | |
|---|---|---|---|
| | **F=256** | **F=512** | **F=768** |
| 1024 | 19.001 | 18.129 | 18.595 |
| 1280 | 17.693 | **16.517** | 18.068 |

Table 2. *Known Categories - Single Hole.* Chamfer Distance results scaled by $10^4$, obtained by changing the auxiliary decoder output and frame dimension.

| **Contrastive Learning Variants** | | **Overall CD** |
|---|---|---|
| Pairs | Batch Size = $98 \times 2$ | 18.030 |
| Quadruplets | Batch Size = $38 \times 4$ | **16.517** |

Table 3. Overall average Chamfer Distance scaled by $10^4$, obtained by changing the Contrastive Learning strategy for the global encoder.

| **Local Denoising Variants** | **Overall CD** | **Parameters** |
|---|---|---|
| DeCo GPDNet [4] | **16.517** | $1.66 \times 10^6$ |
| DeCo DGCNN [5] | 19.667 | $1.13 \times 10^6$ |
| PF-Net [2] | 20.445 | $76.77 \times 10^6$ |

Table 4. Overall average Chamfer Distance scaled by $10^4$, obtained by changing the Denoising Strategy for the local encoder.

## 5. Further Training Details

We provide here more details about the global + local feature aggregation logic. One way to implement the feature combination is by concatenating the global feature vector to each point local feature and feeding them to a 1D conv. layer. In the specific case, the 1D conv. layer has output size 256, which is the dimensionality of the global+local per-point embedding input to the Decoder. This would unnecessarily cause the same global features to be processed $N$ times. We optimized this implementation by separating global and local weight matrices of the 1D conv. layer and combining the obtained representations by summation. This is equivalent to concatenation & conv. but more efficient.

## References


[1] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2

[2] Zitian Huang, Yikuan Yu, Jiawen Xu, Feng Ni, and Xinyi Le. Pf-net: Point fractal network for 3d point cloud completion. In *CVPR*, 2020. 1, 2

[3] Junhyun Lee, Inyeop Lee, and Jaewoo Kang. Self-attention graph pooling. In *ICML*, 2019. 1

[4] Francesca Pistilli, Giulia Fracastoro, Diego Valsesia, and Enrico Magli. Learning graph-convolutional representations for point cloud denoising. In *ECCV*, 2020. 2

[5] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (TOG)*, 2019. 2

[6] Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. Pcn: Point completion network. In *3DV*, 2018. 1

[7] Wenxiao Zhang, Qingan Yan, and Chunxia Xiao. Detail preserved point cloud completion via separated feature aggregation. In *ECCV*, 2020. 1