Denoise and Contrast for Category Agnostic Shape Completion

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

Abstract

In this paper, we present a deep learning model that exploits the power of self-supervision to perform 3D point cloud completion, estimating the missing part and a context region around it. Local and global information are encoded in a combined embedding. A denoising pretext task provides the network with the needed local cues, decoupled from the high-level semantics and naturally shared over multiple classes. On the other hand, contrastive learning maximizes the agreement between variants of the same shape with different missing portions, thus producing a representation which captures the global appearance of the shape. The combined embedding inherits category-agnostic properties from the chosen pretext tasks. Differently from existing approaches, this allows to better generalize the completion properties to new categories unseen at training time. Moreover, while decoding the obtained joint representation, we better blend the reconstructed missing part with the partial shape by paying attention to its known surrounding region and reconstructing this frame as auxiliary objective. Our extensive experiments and detailed ablation on the ShapeNet dataset show the effectiveness of each part of the method with new state of the art results. Our quantitative and qualitative analysis confirms how our approach is able to work on novel categories without relying neither on classification and shape symmetry priors, nor on adversarial training procedures.

1. Introduction

Cameras that scan and render objects in 3D are becoming more and more available as standard feature in many smartphones, drones, robots and cars. Most of these 3D sensing technologies are low-cost stereo cameras as well as depth and laser scanners that output point clouds which are often incomplete due to occlusions, transparency, light reflections or limitations in resolution and viewing angle. The missing regions corrupt the object shape preventing its direct use in tasks like robotic manipulation [28], scene understanding [10], autonomous driving [2] and augmented reality [17]. To overcome those issues, point cloud completion aims at estimating the complete geometry of the missing regions from partial observations.

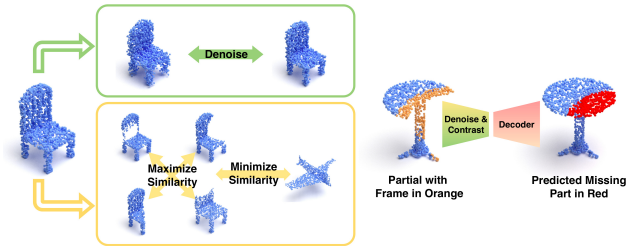

Figure 1. Our DeCo encodes local and global information from the training data via denoising and contrastive learning. The learned embedding is finally decoded to estimate the missing part of the input shape and a frame, *i.e.* a context region around the hole. Thanks to the class-agnostic nature of the self-supervised pretext tasks, our model is effective for point-cloud completion on novel object categories.

There have been several efforts to tackle the completion problem including volumetric representations and related distance fields or mesh models. The most recent literature focuses on the efficient solution of directly inferring new points: a widely used pipeline consists in encoding the partial input into a latent representation which is then decoded to produce the whole shape. However, this strategy leads to an overly difficult setting, where the method attempts at reconstructing the entire point cloud rather than simply filling the missing part. As a consequence, the learned model captures the global geometry more than local properties of each sample, resulting in reconstructions that resemble a generic average object rather than the specific input instance. Naïve design choices of the encoder also contribute to this effect by squashing all the structural information of the point cloud into a single latent global feature with a significant information loss on the details of local regions.

Among the techniques proposed to improve local and

global feature fusion, some try to improve the encoder by describing the point cloud as a collection of surface elements with expansion constraints [16], others model the 3D skeleton of the object [19] or propose new pooling operations [32]. Better decoders have been also developed by revisiting and accumulating low level features as local descriptors [11, 34, 40], adopting a pyramid strategy to recover the missing geometry at multiple resolution levels [11], or including skip and cascaded connections to share information with the encoder [34, 30]. Other approaches exploit local refinements by point upsampling [16, 39], or via adversarial training of patch discriminators [30, 19]. Most of these techniques have never been challenged neither with point clouds corrupted with more than one hole, nor with the reconstruction of object categories unseen at training time. As a matter of fact, in some cases, per-class shape priors are adopted as supervised oracle initialization for the missing points [30]. In order to overcome this closed-set scenario, we propose a novel point cloud completion method that exploits the power of two self-supervised pretext tasks and inherits their category-agnostic properties with a clear generalization effect. Specifically, the main contributions of this work are summarized as follows:

- We propose DeCo (see Figure 1), a model for point cloud completion, that combines local information from Denoising [20] and global information from Contrastive learning [3]. In this way we shed new light on local and global cues which are otherwise reduced just to features at different network depths.
- The self-supervised learned embedding is finally decoded to estimate the missing part of the input shape and a frame, *i.e*. a context region around the hole. This solution avoids the risks of genus-wise distortions [36], and allows to better blend the predicted missing part to the incomplete input.
- DeCo's architecture is designed by exploiting graph convolutions. To the best of our knowledge, the graph logic is used here on point cloud completion for the first time.
- We present extensive experiments on ShapeNet [1] with point clouds corrupted by single and multiple holes as well as testing on novel categories. Our quantitative and qualitative results show the effectiveness of DeCo and set the new state of the art.

2. Related Work

Shape completion has a long tradition in the computer graphics and vision fields and has recently attracted the attention of the deep learning community. Early works engineered effective descriptors by leveraging geometric cues [5, 12] and symmetric priors [27, 18], while data-driven methods were mainly based on retrieval procedures from large 3D shape databases [15, 25]. The most recent learning-based approaches learn a mapping between the partial and corresponding completed shape by exploiting voxel-grids, meshes or point-clouds. Voxel-based approaches exploit 3D convolution networks which lead to large computation and memory cost [4, 8, 24]: this forces a reduction in the resolution of input data and limits the processing of fine-grained shapes. In [7, 29] reference meshes are progressively deformed to match the target. However, this strategy does not generalize across topologies.

Point-cloud representations are much more flexible because new points can be easily added during the learning procedure. The pioneering Point Completion Network (PCN, [39]) was based on an encoder-decoder architecture to reconstruct dense and complete point sets. TopNet [26] includes a tree-structured decoder to improve the point-cloud generation. Differently, Sarmad *et al*. [21] proposed a GAN-based solution where reinforcement learning is used to better control the adversarial loss function. Several works have also revisited both the global feature encoding process and the following local refinement. The method proposed in [34] combines a skip-attention mechanism to avoid information loss about structure details in local regions: the local geometric information is kept when encoding the original incomplete point cloud, and also used at different resolutions in the decoder. In CRN [30], the feature encoder and coarse reconstructor produce a rough complete object shape, which is then updated with points of higher resolution through subsampling from the partial input. Moreover, a feature contraction-expansion unit refines the point position gradually and is further guided by a patch based discriminator, trained to force every local region to have the same pattern as real complete point-clouds. In MSN [16] the refinement procedure is still based on subsampling. The input point cloud and the coarse-grained prediction are recombined to obtain an evenly distributed point cloud, and then a residual model is exploited to enable the generation of fine-grained structures. A different solution based on extracting the 3D skeleton from the partial scan was presented in [19]. The proposed model learns the displacement from the skeletal points to the global surface space. To further preserve fidelity on observable regions, the method also includes local refinement through an adversarial patch discriminator. The approach in [40] tackles the completion problem by processing in a distinct way the partial known shape and its missing chunk. 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. Multi-level features are extracted via a hierarchical learning architecture with gradually increasing grouping radius, inspired by [38].

Very recently PF-Net [11] has shown how to generate exclusively the missing part with good completion perfor-

mance. Multi-scale features are learned from the partial shape to get both local and global information. Then, the missing part is produced hierarchically with primary and secondary points from layers of different depth. Furthermore, an adversarial loss is included to match the distribution of predicted and real missing regions.

As in this last reference, DeCo combines local and global information and focuses mainly on the missing part of the shape. However, we jointly leverage the denoising task to gather local cues and contrastive learning for overall global features. Thus, we extract point features at various scales in a different way with respect to just exploiting the network activations at several depths. Moreover, our solution of involving the context of the missing region as auxiliary reconstruction objective defines a new intermediate framework between the alternatives of reconstructing the entire shape or only the missing part. In this way DeCo ensures a smooth blending of the generated points with the partial input: it takes advantage of the structure around the missing part, while avoiding deformations on the known points.

3. Method

In the following we will indicate with $\boldsymbol{X}_p$ the known partial shape, which is an $N \times 3$ unordered point cloud, and with $\boldsymbol{X}_m$ the corresponding missing part of dimension $M \times 3$, with $M \leq N$. They are respectively the input and output ground truth of our DeCo model. The missing chunk is defined by starting from a random viewpoint and sorting the points in the cloud on the basis of the distance from the observer, finally dropping the closest M set. We will also use $\boldsymbol{X}$ to refer to the original complete shape of dimension $(N + M) \times 3$. To specify each point in the respective clouds we adopt lower-case letters, *e.g.*, $\boldsymbol{x} \in \boldsymbol{X}$. Moreover, we indicate with $\mathbf{Y}$ the generated shape which is composed of $\mathbf{Y}_m$ and $\mathbf{Y}_p$: for the latter it holds $\mathbf{Y}_p = \boldsymbol{X}_p$ since we keep the original partial input while seeking an estimate of the missing part. We train our model starting from a set of $\{\boldsymbol{X}\}_{k=1}^{K}$ complete point clouds which are used both for the pretext tasks that warm-up the encoders, and for the following downstream completion task.

An overview of the proposed DeCo is shown in Figure 2. We will delve into the details of its main components in the next sections. At high level, there are two parallel encoders, implemented as graph convolutional neural network. The local encoder, pre-trained with a denoising task, processes the partial shape to extract a feature vector per input point. The global encoder, pre-trained with contrastive feature learning, produces a single feature vector for the whole point cloud. The two representations are then combined and processed by a graph convolutional decoder, using pooling layers to gradually reduce the number points and match the cardinality of the missing part.

3.1. Local Information by Denoising

Denoising is a highly localized task, mostly relying on low-level geometric cues that are decoupled from the global, high-level semantics and naturally shared over multiple classes. These characteristics perfectly fit with our need of a locality prior for the category-agnostic completion model. We coded the task following [20], which exploits graph convolutional layers in a fully convolutional network. The architecture of the local encoder is shown in the bottom-right part of Figure 2. It is composed of residual blocks that perform graph-based operations to transform the features associated to each point. Specifically, the graph convolution aggregates features belonging to a neighborhood of limited size to maintain locality, while dynamically updating the graph via nearest neighbors in the feature space. With respect to the widely known Dynamic Graph CNN (DGCNN, [31]), the solution in [20] uses a lightweight Edge-Conditioned Convolution (ECC, [22]) layer, well suited for the denoising task. Besides introducing a more general definition of graph convolution, it also addresses the vanishing gradient and over-parametrization issues of the original ECC. Finally, a single graph convolutional layer projects the features back to the 3D space. We drop this last layer after pre-training to retain the high-dimensional feature space in the full DeCo achitecture.

The denoising network is trained by perturbing the input point cloud with additive white Gaussian noise and minimizing the Mean Squared Error (MSE) between the denoised point cloud $\tilde{\boldsymbol{X}}$ and its noiseless ground truth $\boldsymbol{X}$:

$$\mathcal{L}_{MSE} = \frac{1}{N+M} \sum_{\substack{\tilde{\boldsymbol{x}} \in \tilde{\boldsymbol{X}} \\ \boldsymbol{x} \in \boldsymbol{X}}} \|\tilde{\boldsymbol{x}} - \boldsymbol{x}\|^2 \,, \tag{1}$$

with the total loss obtained by averaging the contributions over all the K training samples.

3.2. Global Information by Contrasting

Contrasting positive from negative sample pairs is a common practice for representation and metric learning [33, 23]. The goal is to learn an embedding where similar examples (positive pairs) are mapped close to each other, and dissimilar examples (negative pairs) are mapped far apart. Recently, there has been a shift in the pairs definition, moving from an assignment based on the original sample class label to instance identity [6, 35]. Indeed, it has been shown that treating each sample as a class and exploiting data augmentation to create surrogate data pairs allows to get discriminative features from unlabeled data [3, 9]. Inspired by this literature, we code the global point cloud shape information via contrastive learning and we show the corresponding architecture in the bottom-left part of Figure 2. Specifically, given a randomly sampled mini-batch of point clouds, each one is

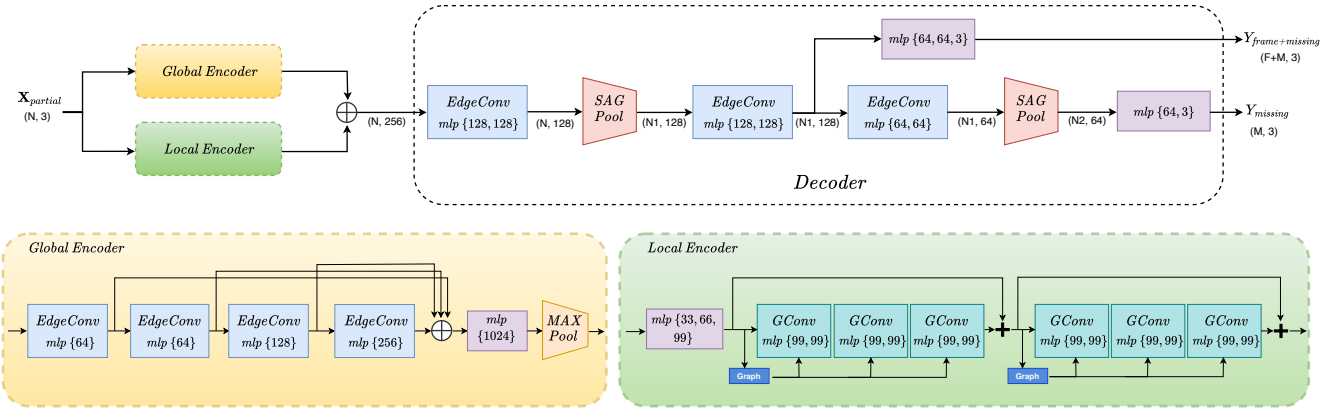

Figure 2. DeCo point cloud completion. Global and local encoders extract semantic and geometric information, respectively, from the partial point cloud by pretraining with contrastive and denoising pretext tasks. The decoder converts this information into the points of the missing part. EdgeConv [31] and GConv [20] are graph convolutional layers, SAG Pool [14] is a graph pooling method. $\bigoplus$ denotes concatenation, + denotes summation. Refer to Sec. 3.4 for all the implementation details.

augmented four times using a combination of rotation (y-axis), random scaling and jittering. All the transformed versions are also randomly cropped. These new variants $\{\boldsymbol{X}_{4k}, \boldsymbol{X}_{4k-1}, \boldsymbol{X}_{4k-2}, \boldsymbol{X}_{4k-3}\}$ are provided as input to four stacked EdgeConv blocks [31]. The obtained convolved representations at different depths are concatenated and further processed by an MLP layer. The global shape feature vector is finally obtained by performing max pooling. A further MLP head (two hidden layer inter spaced by ReLU) is used to project the global shape feature vector to a lower dimensional space (128). This yields a representation per sample variant $\mathcal{P}(k) = \{\boldsymbol{z}_{4k}, \boldsymbol{z}_{4k-1}, \boldsymbol{z}_{4k-2}, \boldsymbol{z}_{4k-3}\}$ that enters a generalized version of the the Normalized Temperature-scaled cross entropy loss (NT-Xent):

$$\mathcal{L}_{NT-Xent} = \frac{-1}{|\mathcal{P}(i)|} \sum_{j \in \mathcal{P}(i)} \log \frac{\exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{4K} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_k)/\tau)} . \tag{2}$$

Here $\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)$ is the cosine similarity between two feature vectors, and the final loss is computed across all positive pairs within a quadruplet, while considering, as negatives all the remaining transformed samples of the mini-batch. Following standard practice [3], we include the MLP projection head only during the global encoder pre-training, while it is removed in the final DeCo architecture.

This task is well suited for our global encoder: it promotes a robust semantic embedding by learning instance representations that are close to each other regardless of which portion of the point cloud is missing, and thus capturing a global understanding of it. Critically, it does not require supervision in the form of class labels.

3.3. Framing and Reconstructing the Missing Part

The information collected by the local and global encoders are finally combined to guide the generation of the missing shape part. The two obtained feature embeddings are aggregated and fed as input to the following decoder network. Our decoder is composed by three EdgeConv layers [31] and two Self-Attention Graph Pooling layers (SAG Pool, [14]), whose purpose is to reduce the number of points down to the number of points of the missing part. As can be noticed from the top part of Figure 2 the decoder has two outputs at different levels. The final head is defined by an MLP that generates $\mathbf{Y}_m$. The intermediate head positioned between the two central EdgeConv layers steers the feature space to correctly represent the region around the missing part as well as the missing part itself.

Specifically, starting from the sorted points used for the definition of the missing chunk $\boldsymbol{X}_m$, we extend our attention to the following set of F point in the same list to define the *frame + missing region* $\boldsymbol{X}_{fm}$ of dimension $(F+M)\times 3$. We regularize training by constraining the decoder to generate an estimate of the missing part $\mathbf{Y}_m$ consistent with the ground truth $\boldsymbol{X}_m$, and to reconstruct correctly the frame and the missing part as $\mathbf{Y}_{fm}$ from the intermediate head. For both objectives, the training procedure minimizes the Chamfer Distance (CD) loss:

$$\mathcal{L}_{CD} = \frac{1}{2M}\Big\{ \sum_{\boldsymbol{x} \in \boldsymbol{X}_m} \min_{\boldsymbol{y} \in \mathbf{Y}_m} \|\boldsymbol{x}-\boldsymbol{y}\|_2^2 + \sum_{\boldsymbol{y} \in \mathbf{Y}_m} \min_{\boldsymbol{x} \in \boldsymbol{X}_m} \|\boldsymbol{y}-\boldsymbol{x}\|_2^2 \Big\}$$
$$+ \frac{1}{2(M+F)}\Big\{ \sum_{\boldsymbol{x} \in \boldsymbol{X}_{fm}} \min_{\boldsymbol{y} \in \mathbf{Y}_{fm}} \|\boldsymbol{x}-\boldsymbol{y}\|_2^2 + \sum_{\boldsymbol{y} \in \mathbf{Y}_{fm}} \min_{\boldsymbol{x} \in \boldsymbol{X}_{fm}} \|\boldsymbol{y}-\boldsymbol{x}\|_2^2 \Big\} . \tag{3}$$

During testing we do not have control on the exact nature of the missing shape part and the output of the intermediate head is neglected.

3.4. Implementation Details

We designed the *local encoder* architecture on the basis of a graph constructed by searching dynamically for the k-

nearest neighbor ($k = 8$) of each point in terms of Euclidean distance between their feature vectors. For pre-training we used shapes corrupted by Gaussian noise with average set to 0 and standard deviation equal to 0.02. The *global encoder* has four stacked EdgeConv layers, each with $k = 24$ nearest neighbors. For pre-training we used shapes with random crops of 25% (512 points out of point clouds of $N = 2048$). We set the temperature scaling parameter in the NT-Xent loss $\tau = 0.5$. The *decoder* alternates EdgeConv layers with $k = 16$ and two attention-based pooling layers, respectively based on graphs with $k = 16$ and $k = 6$ neighbors. The intermediate feature dimensions indicated in Figure 2 are $N_1 = 1280$ and $N_2 = 512$ with $M = F = 512$.

We trained one single network over all 13 known object categories for 240 epochs with a batch size of 30. We used Adam [13] with initial learning rate set to 0.001, halved every 25 epochs for the (pre-trained) encoder and every 40 epochs for the decoder. We implemented all the network using PyTorch and train it on two NVIDIA Titan RTX GPUs (PyTorch DataParallel) with CUDA 10.0. The model is finally tested on a single GPU with batch size 64. Ablation experiments ran on hpc cluster with NVIDIA V100 GPUs.

The code of DeCo is available at https://github.com/antoalli/Deco.

4. Experiments

4.1. Dataset, Baselines and Evaluation Metric

To evaluate the proposed DeCo we follow the experimental protocol of [11], selecting 13 object classes from the benchmark dataset Shapenet-Part [37]. The total number of shapes is 14473 with 11705 used for training and 2768 for testing. All the input point cloud data is centered at the origin and their coordinate are normalized to $[-1, 1]$. The ground truth is created by sampling 2048 points uniformly on each shape. For the novel categories we selected 12 objects classes from Shapenet-Core [1], getting a total of 7873 shapes. In the chosen set, six classes are semantically related to the seen ones (bicycle-motorbike, basket-bag, helmet-cap, bowl-mug, rifle-pistol, vessel-airplane), while the remaining six (piano, bookshelf, bottle, clock, microwave, telephone) were chosen randomly.

We compare against several recent completion methods: PCN [39], MSN [16], CRN [30], as well as two variants of PF-Net [11] with and without (vanilla version) its adversarial discriminator. For all these baselines we ran the code provided by the authors to get both the quantitative and qualitative results. To keep a fair comparison on 2048 points, for CRN we consider a single iteration through its refinement sub-network. We quantitatively assess the performance of all the methods by using the Chamfer Distance (CD) on the reconstructed missing part. More precisely, we follow the evaluation strategy already validated by PF-Net:

Category	PCN [39]	MSN [16]	CRN [30]	PF-Net vanilla [11]	PF-Net [11]	DeCo
Airplane	31.515	15.907	39.334	11.015	10.805	**10.003**
Bag	37.825	59.185	33.593	40.000	38.485	**28.508**
Cap	66.275	40.276	53.146	49.945	50.450	**36.436**
Car	24.320	24.176	39.537	21.925	**21.640**	22.963
Chair	31.265	20.751	28.688	19.130	19.490	**16.428**
Lamp	93.745	41.094	30.207	41.555	42.910	**24.150**
Laptop	22.460	11.718	26.393	11.520	**11.220**	12.706
Motorbike	34.420	21.276	41.292	20.525	19.905	**19.136**
Mug	35.905	57.007	41.153	32.800	**31.880**	34.239
Pistol	29.490	14.560	26.845	11.395	**10.885**	12.266
Skateboard	23.815	14.146	34.358	12.275	12.365	**9.861**
Table	24.775	22.103	23.953	20.560	20.845	**17.120**
Guitar	10.540	6.959	15.224	4.350	**4.425**	4.482
Overall	34.095	22.410	29.044	20.209	20.445	**16.517**

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

Method	Single Hole 25%	Single Hole 50%	Two Holes
PF-Net vanilla	20.209	20.950	25.140
PF-Net	20.445	19.325	33.632
DeCo	**16.517**	**17.554**	**24.430**

Table 2. *Known Categories - Robustness Test* Overall average Chamfer Distance scaled by 10^4. The results confirm the advantage of DeCo against its best competitor PF-Net.

for the methods predicting the overall shape we report the CD computed only on the M closest points to the crop centroid. We present the results per class and the overall average CD on all the test shapes.

4.2. Known Categories

Quantitative Analysis Table 1 presents the completion results that indicate the superiority of DeCo with respect to all the considered baseline approaches. More in details, DeCo outperforms its best competitor PF-Net on eight out of thirteen categories, with a large margin on lamp, cap, and bag.

Qualitative Analysis Figure 3 shows the point cloud reconstructed by the different baseline methods and by DeCo. On the airplane point cloud, most of the baselines lack one or both the wing engines. On the chair point cloud, DeCo is the only method to reconstruct the missing leg without any significant noise. The guitar highlights the clear advantage of generating only the missing part, rather than reconstructing the whole shape, while also showing how DeCo is much more precise than the two PF-Net variants. For the table, all the baselines present artifacts either on the horizontal surface or on the legs. Finally, the last row shows a failure case: none of the methods is able to generate a precise reconstruction for the missing part of the lamp. In general, it is evident that the results of PCN are too noisy while those of MSN are often discontinuous or incomplete.

Robustness Test To analyze the robustness of DeCo we ran two sets of experiments over all the known classes by considering a single larger hole or two separate smaller holes

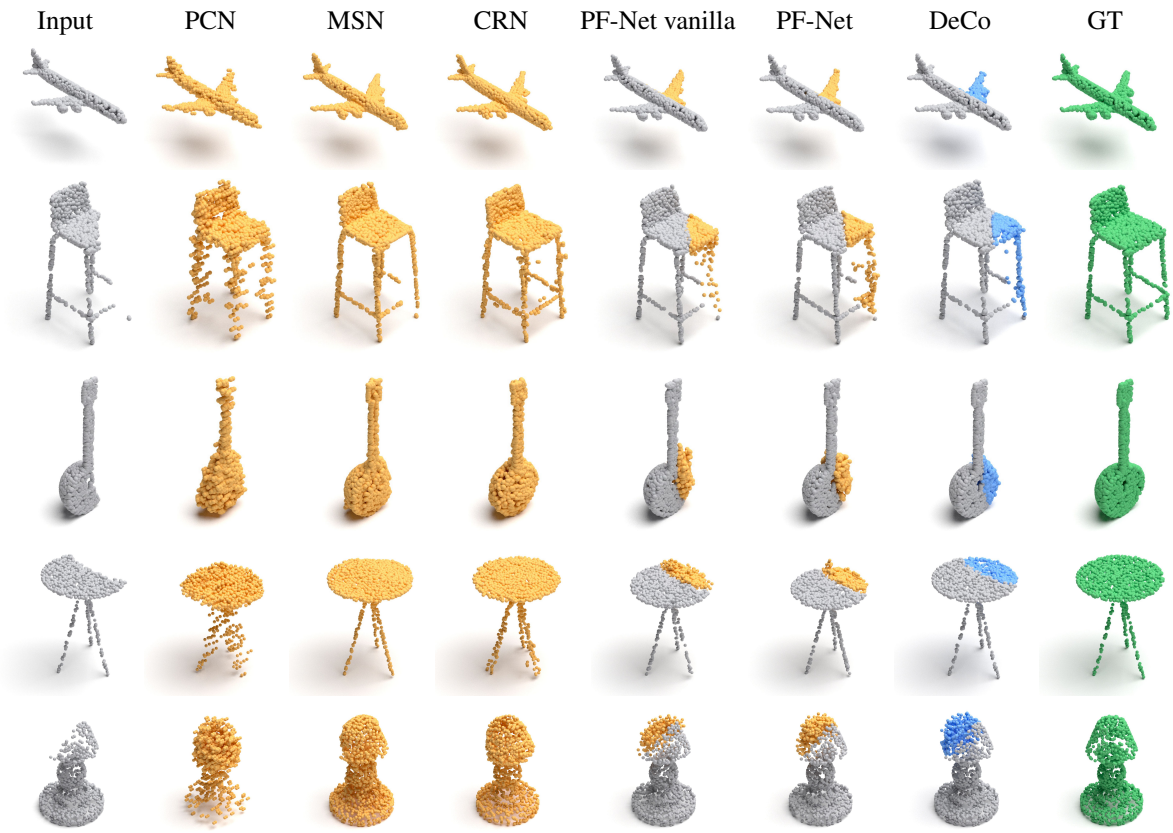

Figure 3. *Known Categories - Qualitative*. The first four rows (airplane, chair, guitar, table) show how DeCo generates the missing shape part with more details and a less noisy appearance than its competitors. The last row (lamp) shows a general failure case for all the approaches. For PFNet and DeCo we visualize the predicted missing part (resp. yellow and blue points) w.r.t. the partial input (grey).

in the point clouds. In the first case, we changed M passing from 512 to 1024, thus extending the missing part from 25% to 50% of the original shape. In the second case, we randomly chose two viewpoints, each used to define the origin of a 12.5% ($M_1 = M_2 = 256$ points) hole. We focus on the comparison with the best performing baseline, *i.e.* PF-Net, and the results in Table 2 confirm that DeCo outperforms it in all the settings, thus showing a stronger robustness. It is interesting to note that, when dealing with two holes, the adversarial discriminator of PF-Net is detrimental: by monitoring the training, our intuition is that the PF-Net reconstructed output remains too different from the ground truth to properly trigger the beneficial effect of the adversarial game.

4.3. Novel Categories

Quantitative Analysis We extend our experimental analysis to novel categories, unseen at training time. Given its low reconstruction accuracy, we disregard PCN here while keeping all the other baselines. The quantitative results in Table 3 show how DeCo outperforms all the considered competitors by a large margin regardless of the semantic relatedness between the known and new classes.

Categories	MSN [16]	CRN [30]	PF-Net vanilla [11]	PF-Net [11]	DeCo
Similar					
Bicycle	47.423	64.275	49.779	47.186	39.684
Basket	48.100	50.692	58.866	57.066	**34.613**
Helmet	71.161	57.851	63.742	69.849	**47.412**
Bowl	52.002	63.357	97.316	78.793	**35.209**
Rifle	34.712	47.239	25.438	28.684	**12.004**
Vessel	30.948	41.418	27.122	31.114	**18.836**
Overall	35.544	46.166	31.232	33.844	**17.680**
Dissimilar					
Piano	62.969	61.643	62.131	62.994	**49.429**
Bookshelf	48.397	44.738	58.920	55.123	**34.681**
Bottle	29.580	20.134	25.543	24.578	**20.002**
Clock	57.222	38.132	50.964	48.373	**32.826**
Microwave	53.354	56.259	61.702	56.152	**41.877**
Telephone	38.032	25.554	38.085	32.063	**20.106**
Overall	45.049	34.625	45.014	41.449	**28.403**

Table 3. *Novel Categories - Quantitative*. Chamfer distance on the missing region of point clouds scaled by 10^4. The lower, the better.

Qualitative Analysis Table 4 collects the point clouds completed by the different considered methods. In general DeCo is the only approach that, besides not loosing infor-

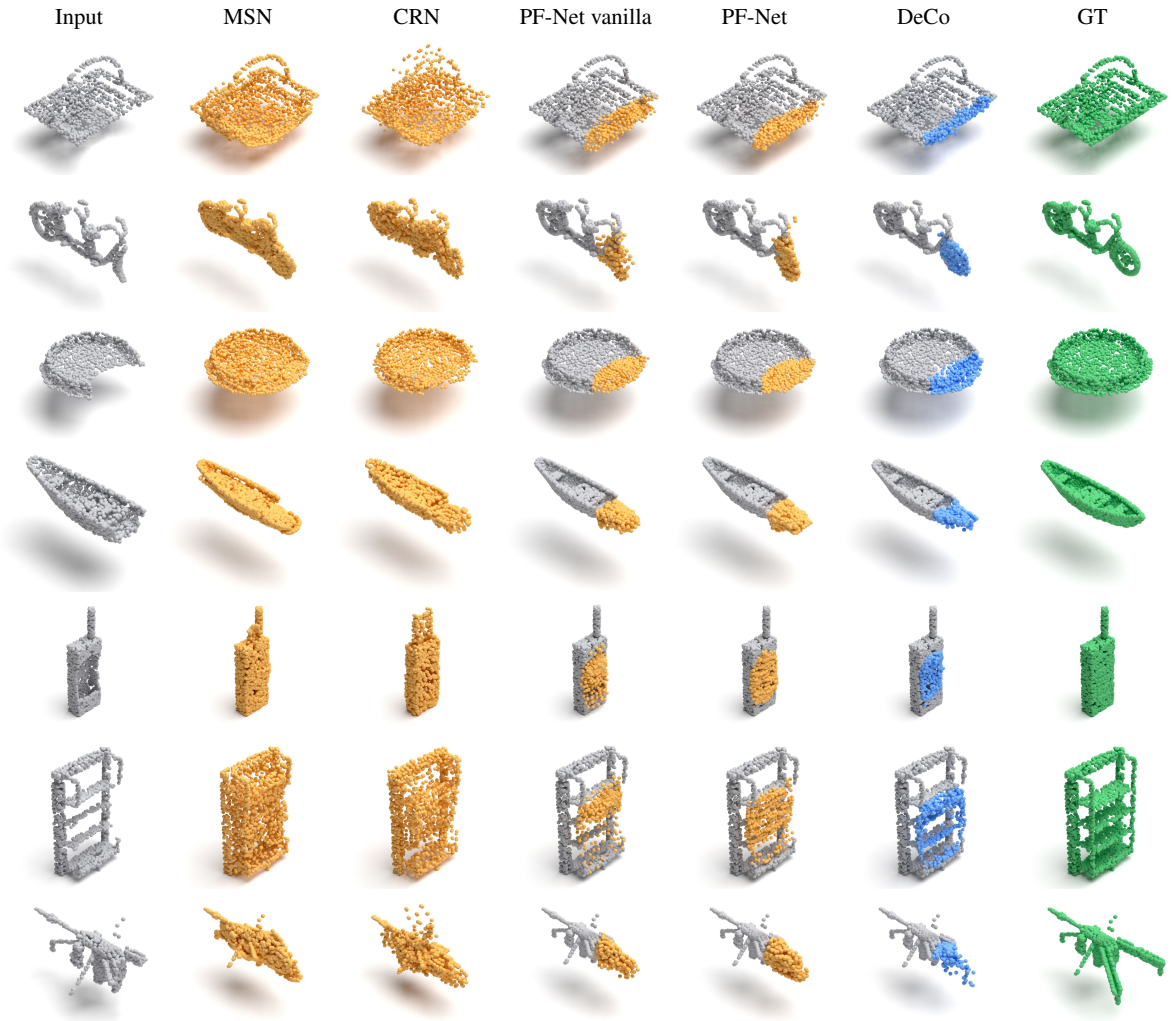

Figure 4. *Novel Categories - Qualitative*. Completion results for samples of class basket, bicycle, bowl, vessel, telephone, bookshelf and rifle (top to bottom row).

mation on the partial input, is able to fill the hole maintaining a smooth transition to the missing part as well as shape continuity (*e.g*. the boarder of the bowl, the pointy end of the vessel). The results of MSN and CRN are often noisy (*e.g*. basket and bicycle) or present artifacts (*e.g*. telephone), while the PF-Net variants are less precise than DeCo. The bookshelf can be considered a mild failure case: none of the approaches is able to complete correctly the second and third partially missing shelves. DeCo has the best overall appearance, also considering the details of the vertical shelf connections, but the second and first shelf get merged together. The failure is even more evident in the case of rifle in the last row.

4.4. Ablation Study

We can identify three main components in DeCo: the local encoder, the global encoder and the auxiliary condition of reconstructing the frame region around the missing part. To carefully study the effect of each of them we perform extensive ablation experiments and organize Table 4 into three groups. The first and second groups analyze the benefits induced by the pretext tasks (denoising and contrastive learning) when respectively the frame spatial constraint is turned off and on; in the third group we consider at the global encoder a supervised classification pretext for comparison with our contrastive formulation.

If we focus on the known classes, the first row in Table 4 indicates that, by turning off all the pre-trainings and the frame constraint, we get a CD (23.865) similar to MSN (22.410) but higher than what obtained by PF-Net (20.445). This indicates at the same time a good backbone design for DeCo, as well as clear room for improvement in the training procedure. For the unknown classes our basic architecture outperforms (26.811) the competitors when dealing with novel classes similar to the known ones (best case: PF-Net Vanilla 31.232), but it is significantly worse (40.419) in

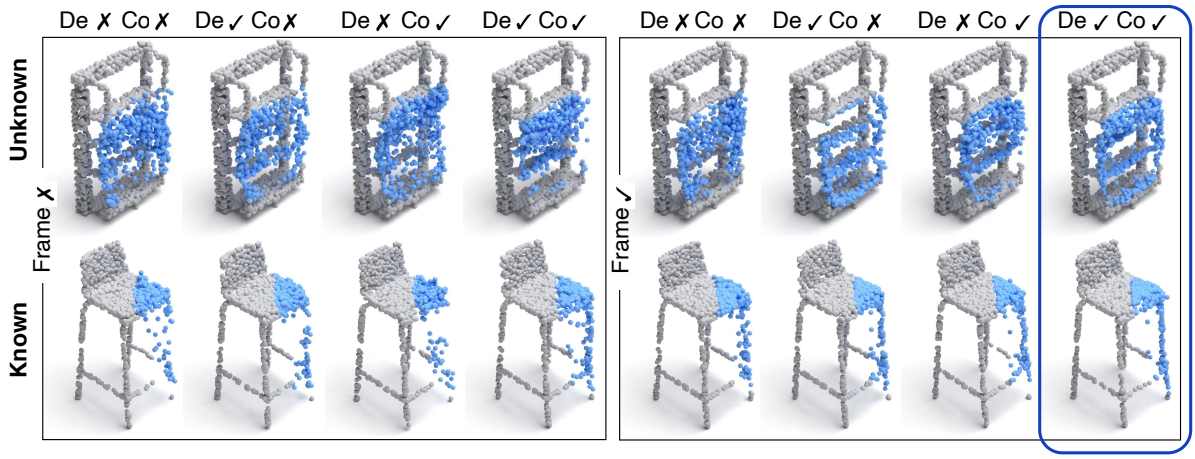

Figure 5. *Visual Ablation*. Qualitative results showing the impact of each single component of our model. First row: Unknown object Bookshelf. Second row: Known object Chair. Given the partial input points (grey), DeCo predicts the missing part points (blue).

case of dissimilar ones (best case: CRN 34.625). This confirms that the backbone per-se is not able to generalize.

The following rows in the first part of the table show how both local denoising and global contrastive pre-training lead to lower reconstruction errors. The former appears more effective than the latter: indeed local information is extremely relevant to reconstruct the shape details. Still, their combination always produce a further accuracy gain, obtaining already state of the art results (DeCo known/unknown sim./unknown diss.: (18.742 / 19.364 / 32.945) vs (PF-Net 20.445 / PF-Net van. 31.232 / CRN 34.625)).

The second group of results in the table highlights the effect of including the frame regularization: it provides an evident performance uplift, validating our hypothesis of better blending the reconstructed points with the existing ones.

Finally, the bottom part of the table presents the effect of substituting our unsupervised contrastive pretext with a more informed supervised one. Although this choice appears valuable and allows to outperforms the top competitors, the obtained reconstruction error results are worse than what obtained by DeCo. The difference is particularly evident on the unknown classes where the information coming from the closed-world classification task is clearly unable to support generalization. Our unsupervised local and global pretext tasks provide less distortions in the generated missing part besides not requiring costly labeled data. The effect of each component of our model is also shown by the visual ablation in Figure 5.

Local	**Global**		**Frame**	**Overall**		
Denoise	Cla.	Contr.		Known	Unknown Sim.	Unknown Diss.
✗	✗	✗	✗	23.865	26.811	40.419
✓	✗	✗	✗	21.022	22.213	33.602
✗	✗	✓	✗	23.067	25.723	36.567
✓	✗	✓	✗	18.742	19.364	32.945
✗	✗	✗	✓	20.586	22.538	32.661
✓	✗	✗	✓	18.131	20.064	31.824
✗	✗	✓	✓	18.995	20.989	32.070
✓	✗	✓	✓	**16.517**	**17.680**	**28.403**
✗	✓	✗	✗	24.364	29.470	40.269
✓	✓	✗	✗	20.272	21.187	34.592
✗	✓	✗	✓	19.378	21.942	32.456
✓	✓	✗	✓	18.699	21.712	31.645

Table 4. *Ablation analysis*. Chamfer Distance scaled by 10^4. The results show the gain provided by each single component of our model. At the global encoder, we also compare the Contrastive pretext (*Contr.*) with a supervised Classification pretext (*Cla.*).

5. Conclusions

In this work we introduced a new point cloud completion method that encodes shape knowledge via two self-supervised pretext task: denoising to gather local cues and contrastive learning for global information. Our DeCo focuses on reconstructing the missing part of the point cloud by also exploiting a context frame region as anchor reference: it avoids to re-generate the whole shape while keeping strong spatial continuity with the observed partial input. The obtained completion results as well as the conducted ablation and robustness studies indicate that DeCo outperforms existing competitors defining the new state of the art on the standard closed-class setting. Moreover we extensively evaluated DeCo on novel categories, further showing the effectiveness of our approach.

How to deal with fine-grained structured objects as rifles or modern design lamp and bookshelves remains a challenging open question for all point cloud completion methods. For the future we plan to extend DeCo in this direction as well as on real-world scans where the missing part issue comes along with extremely sparse partial inputs.

Acknowledgements This work was partially supported by the CHIST-ERA BURG Project (TT). We also acknowledge that the research activity herein was carried out using the IIT HPC infrastructure.

References

[1] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015. 2, 5

[2] Siheng Chen, Baoan Liu, Chen Feng, Carlos Vallespi-Gonzalez, and Carl Wellington. 3d point cloud processing and learning for autonomous driving. *IEEE Signal Processing Magazine*, May 2020. 1

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2, 3, 4

[4] Angela Dai, Charles Ruizhongtai Qi, and Matthias Nießner. Shape completion using 3d-encoder-predictor cnns and shape synthesis. In *CVPR*, 2017. 2

[5] James Davis, Stephen R. Marschner, Matt Garr, and Marc Levoy. Filling holes in complex surfaces using volumetric diffusion. In *3DPVT*, 2002. 2

[6] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *NIPS*, 2014. 3

[7] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In *CVPR*, 2018. 2

[8] X. Han, Z. Li, H. Huang, E. Kalogerakis, and Y. Yu. High-resolution shape completion using deep neural networks for global structure and local geometry inference. In *ICCV*, 2017. 2

[9] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 3

[10] Ji Hou, Angela Dai, and Matthias Nießner. 3d-sis: 3d semantic instance segmentation of rgb-d scans. In *CVPR*, 2019. 1

[11] Zitian Huang, Yikuan Yu, Jiawen Xu, Feng Ni, and Xinyi Le. Pf-net: Point fractal network for 3d point cloud completion. In *CVPR*, 2020. 2, 5, 6

[12] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (TOG)*, 32(3), 2013. 2

[13] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *ICLR*, 2015. 5

[14] Junhyun Lee, Inyeop Lee, and Jaewoo Kang. Self-attention graph pooling. In *ICML*, 2019. 4

[15] Yangyan Li, Angela Dai, Leonidas Guibas, and Matthias Nießner. Database-assisted object retrieval for real-time 3d reconstruction. In *Computer Graphics Forum*. Wiley Online Library, 2015. 2

[16] Minghua Liu, Lu Sheng, Sheng Yang, Jing Shao, and Shi-Min Hu. Morphing and sampling network for dense point cloud completion. In *AAAI*, 2020. 2, 5, 6

[17] Weiquan Liu, Baiqi Lai, Cheng Wang, Xuesheng Bian, Wentao Yang, Yan Xia, Xiuhong Lin, Shang-Hong Lai, Dongdong Weng, and Jonathan Li. Learning to match 2d images and 3d lidar point clouds for outdoor augmented reality. In *IEEE VR*, 2020. 1

[18] Niloy J. Mitra, Leonidas J. Guibas, and Mark Pauly. Partial and approximate symmetry detection for 3d geometry. *ACM Transactions on Graphics (TOG)*, 25(3):560–568, 2006. 2

[19] Yinyu Nie, Yiqun Lin, Xiaoguang Han, Shihui Guo, Jian Chang, Shuguang Cui, and Jian Jun Zhang. Skeleton-bridged point completion: From global inference to local adjustment. In *NIPS*, 2020. 2

[20] Francesca Pistilli, Giulia Fracastoro, Diego Valsesia, and Enrico Magli. Learning graph-convolutional representations for point cloud denoising. In *ECCV*, 2020. 2, 3, 4

[21] Muhammad Sarmad, Hyunjoo Jenny Lee, and Young Min Kim. Rl-gan-net: A reinforcement learning agent controlled gan network for real-time point cloud shape completion. In *CVPR*, 2019. 2

[22] Martin Simonovsky and Nikos Komodakis. Dynamic edge-conditioned filters in convolutional neural networks on graphs. In *CVPR*, 2017. 3

[23] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, 2016. 3

[24] David Stutz and Andreas Geiger. Learning 3d shape completion under weak supervision. *IJCV*, 128(5):1162–1181, 2018. 2

[25] Minhyuk Sung, Vladimir G. Kim, Roland Angst, and Leonidas Guibas. Data-driven structural priors for shape completion. *ACM Transactions on Graphics (TOG)*, 34(6), 2015. 2

[26] Lyne P Tchapmi, Vineet Kosaraju, S. Hamid Rezatofighi, Ian Reid, and Silvio Savarese. Topnet: Structural point cloud decoder. In *CVPR*, 2019. 2

[27] Sebastian Thrun and Ben Wegbreit. Shape from symmetry. In *ICCV*, 2005. 2

[28] Jacob Varley, Chad DeChant, Adam Richardson, Joaquín Ruales, and Peter K. Allen. Shape completion enabled robotic grasping. In *IROS*, 2017. 1

[29] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *ECCV*, 2018. 2

[30] Xiaogang Wang, Marcelo H. Ang Jr. , and Gim Hee Lee. Cascaded refinement network for point cloud completion. In *CVPR*, 2020. 2, 5, 6

[31] 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. 3, 4

[32] Yida Wang, David Joseph Tan, Nassir Navab, and Federico Tombari. Softpoolnet: Shape descriptor for point cloud completion and classification. In *ECCV*, 2020. 2

[33] Kilian Q. Weinberger and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.*, page 207–244, 2009. 3

[34] Xin Wen, Tianyang Li, Zhizhong Han, and Yu-Shen Liu. Point cloud completion by skip-attention network with hierarchical folding. In *CVPR*, 2020. 2

[35] Zhirong Wu, Yuanjun Xiong, X Yu Stella, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018. 3

[36] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *CVPR*, 2018. 2

[37] Li Yi, Vladimir G. Kim, Duygu Ceylan, I-Chao Shen, Mengyan Yan, Hao Su, Cewu Lu, Qixing Huang, Alla Sheffer, and Leonidas Guibas. A scalable active framework for region annotation in 3d shape collections. *ACM Trans. Graph.*, 35(6), Nov. 2016. 5

[38] Lequan Yu, Xianzhi Li, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. Pu-net: Point cloud upsampling network. In *CVPR*, 2018. 2

[39] Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. Pcn: Point completion network. In *3DV*, 2018. 2, 5

[40] Wenxiao Zhang, Qingan Yan, and Chunxia Xiao. Detail preserved point cloud completion via separated feature aggregation. In *ECCV*, 2020. 2