Towards a Density Preserving Objective Function for Learning on Point Sets

Haritha Jayasinghe and Ioannis Brilakis

University of Cambridge

Abstract. Accurate measurement of the discrepancy between point sets is crucial for point cloud learning tasks. Chamfer distance (CD) is favoured over more effective loss metrics such as Earth Mover's Distance (EMD) for this purpose due to its computational efficiency. Previous investigations into loss function improvements almost exclusively focus on 3D losses as static metrics, and ignore their dynamic behaviour during training. We show that directly modifying the correspondence criteria can prevent clustering of points during training, and lead to more uniform point distributions. We propose UniformCD, a novel 3D distance metric that prioritises matching the relative local densities of point neighbourhoods when assigning correspondences. Experiments demonstrate that the proposed loss function improves performance on a variety of tasks such as cloud completion, parametric model optimisation, as well as downstream task performance when used for self-supervised learning, achieving SOTA EMD results among point set objective functions. We show that the proposed method exploits local density information to converge towards globally optimum density distributions, narrowing the disparity between CD and EMD. Source code will be publicly released.

1 Introduction

The advent of deep learning, combined with the widespread popularity of 3D scanning devices has led to significant work on point cloud learning tasks. These include point cloud completion [30, 33, 37, 39], denoising [17, 29, 40], up-sampling [15, 41] and generation [10, 25, 35]. All of these tasks are heavily reliant on the ability of a network to learn an accurate representation of a point cloud. Learning such a representation is, in turn, reliant on accurately measuring the distance between a generated and input point cloud. In addition to learning representations, such a measurement is also crucial for iterative optimisation tasks involving point clouds such as registration and parametric optimisation.

A distance computation between two point sets can be devolved into three stages. First, correspondences between individual points are identified. Next, the distance between each corresponding set of points is measured. Finally, these distances are aggregated to produce a single distance value.

The underlying intent for any point cloud distance metric is to provide a comparison between the underlying geometric surfaces that generate each of the two point clouds, known as their 'supports'. A given point cloud is assumed

to have been sampled from this support. Thus, the distance between the two point sets acts as a proxy for the distance between their supports. This makes the comparison of distance metrics particularly difficult as there is no 'true' ground truth measure for the discrepancy between supports. In accordance with Goodheart's law, whenever a generative model is trained with a certain distance metric as a loss, it tends to outperform models trained on other losses at that particular metric. Consequently, visual comparisons of generated points trained on various distance metric are often used to evaluate distance metrics.

CD and EMD are the two most commonly used point set distance metrics. EMD is generally accepted to be more effective, and produces more visually consistent results [18, 31]. However, CD is commonly used in practice as it is much more computationally efficient. In particular, CD tends to converge to sub-optimal solutions when used as an objective function [12]. The performance gap between CD and EMD has inspired a number of alternative distance functions [4, 11, 12, 18] that seek to improve the performance of CD.

In this work, we show that the requirements of a distance function vary based on its use case. To explore these requirements, aside from their well-investigated static properties, we examine the dynamic properties 3D losses demonstrate when they are used as objective functions. Consequently, we propose a novel modification aimed at improving the convergence of CD when used as a loss function.

Our proposed method improves convergence by reducing the likelihood of multiple points in one cloud corresponding to a single point in the other, resulting in local minima (Figure 1). This is performed by modifying the correspondence criteria to take the local density of a corresponding point into account, discouraging matches with points within dense neighbourhoods. Our method maintains the same computational complexity as CD ($\mathcal{O}(n \log n)$). We demonstrate that our method outperforms CD and alternative distance metrics on completion, parametric optimisation, and self-supervised learning in EMD metrics. We further show that our method is agnostic to choice of model architecture.

In summary this research provides the following contributions;

- We present a novel analysis of requirements of a point set distance metric used for optimisation;
- Based on insights from this analysis, we propose a novel distance metric which prioritises uniform density distributions;
- We provide evaluations to illustrate the effectiveness of our proposed metric and demonstrate that it lessens the gap between CD and EMD.

2 Background

2.1 Point Set Distance Metrics

We identify two primary use cases for measuring the discrepancy between a pair of point clouds. The first is as as an **objective function**. Consider training a generative model which learns a representation of a point cloud. 3D discrepancy

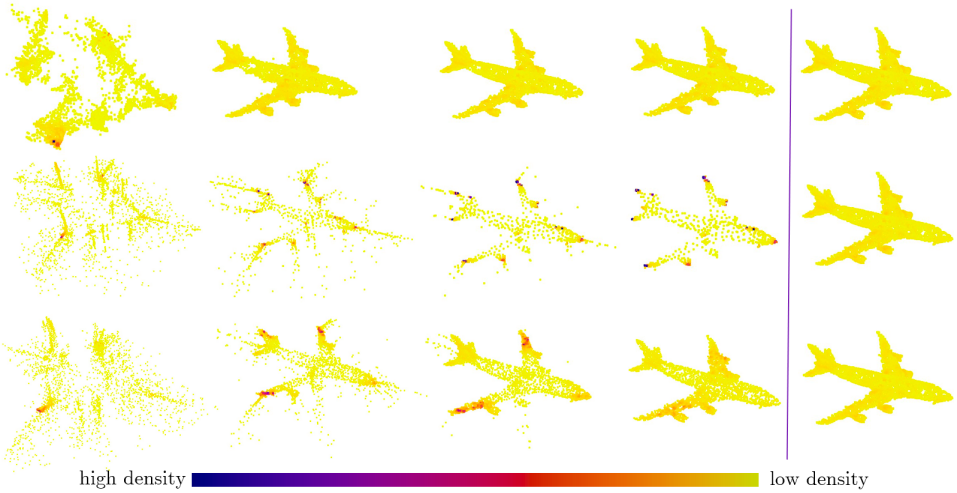

Fig. 1: Morphing a sphere into a given shape (left → right) by minimising Earth Mover's Distance (top), chamfer distance (middle) and UniformCD (bottom). The ground truth is given on the right. Note the occurrence of high density clusters on CD, resulting in very low density in other regions. UniformCD significantly reduces this effect, resulting in a more uniform cloud, similar to EMD.

serves as the loss function for this training. The model attempts to minimise the distance between a ground truth point set known as the target, and a point set generated by the model known as the source. This method is used to train autoencoder architectures for completion of occluded point clouds [39, 42], denoising [22,40] and resampling [1,36]. It is also increasingly utilised for self-supervised learning on point clouds, in order to improve performance on downstream tasks such as semantic segmentation [27,28,38]. Aside from training neural nets, these objective functions are also used for iterative optimisation tasks. Here, the goal is to directly optimise a set of parameters which leads to the generation or manipulation of a point set. Point cloud registration [6,13] and parametric modelling [7, 23] are common examples of this process.

When used as an objective function, the metric must be computationally efficient as it is repeatedly calculated during optimisation. The discrepancy must also be differentiable to enable gradient descent optimisation via backpropagation. Furthermore, coverage of all points in the target cloud is desirable in a discrepancy metric. In essence, this equates to a surjective mapping from the source points to the target points and vice versa. Low coverage leads a model to generate multiple points that cluster towards a single point in the target cloud [12] and ignore other points. This results in the loss metric reaching a local minima. Point clouds generated by such models show large density variations compared to the ground truth, similar to the second row of Figure 1. This is a dynamic property; it is not observed when the metric is used in a single iteration, but emerges over time when a metric is repeatedly utilised for optimisation.

Variation of point correspondences over iterations is another dynamic property. Iterative optimisation using gradient descent benefits from a smooth loss function. However, if the set of corresponding point pairs change significantly between two iterations, the resultant loss may be difficult to optimise. Additionally, invariance to noise is also desirable in an objective function.

The second use case is simply as an **evaluation metric** for gauging the effectiveness of the above tasks [10,15,33,34,39]. An evaluation metric has notably different requirements in comparison to the previous use case. Here, accuracy, i.e. the ability of the metric to represent the true deviation between the supports of two point sets, is preferable over computational efficiency. Differentiability and smoothness are no longer required. Coverage is still desirable, however convergence towards local minima is no longer a concern. Thus, we find that the requirements of a point set discrepancy metric differ significantly based on its use case.

With these requisites in mind, let us examine existing point set distance metrics.

2.2 Chamfer Distance and Earth Mover's Distance

Chamfer discrepancy (CD) is the most commonly used distance metric by a significant margin. It is computed by summing the squared distance between each point and its nearest neighbour in the other cloud. It has proven to be effective in practice, and has low computational complexity. When nearest neighbour search is performed using an octree structure, CD has time complexity $\mathcal{O}(n \log n)$ [20], where n is the number of points. for a pair of point clouds X and Y, bidirectional CD is given by,

$$D_{CD} = \sum_{j} \min_{i} ||x_i - y_j||_2 + \sum_{i} \min_{j} ||x_i - y_j||_2 \tag{1}$$

The second most common distance metric is Earth Movers Distance (EMD). EMD seeks a bijection, $F(X)$ from X to Y, which minimises the squared distance between each point x and its corresponding point $F(x)$ in Y. Searching for this mapping is an iterative process with complexity $\mathcal{O}(n^3 \log n)$ [24]. Unlike CD, both clouds must have an equal number of points for EMD calculation. Intuitively, EMD measures the distance that each point in X must be moved in order to convert X into Y, such that the total distance of movement is minimised. EMD is given by;

$$D_{EMD} = \min_{F:X \to Y} \sum_{i} ||x_i - F(x_i)||_2 \tag{2}$$

Previous work provides comparisons between CD and EMD. It has been qualitatively and quantitatively demonstrated that EMD optimisation produces more consistent results than CD [18, 31]. Nguyen et al [18] additionally argue that divergence when optimised with CD is weaker than with EMD. This infers that minimising EMD guarantees that CD is minimised, while the opposite is not guaranteed. However, we note that this inference is contingent upon the

existence of an optimum 1 : 1 mapping between all points of the two clouds, which is not guaranteed. For instance, when registering two clouds, occlusions and noise will cause variations between clouds [6]. In such cases, EMD diverges to a sub-optimal solution by seeking a 1 : 1 mapping.

Optimisation with EMD prevents clustering as the mapping is injective, i.e. multiple points in the source cannot correspond to a single point in the target. This results in a balanced density distribution in comparison to CD. As a metric, EMD has been demonstrated to be superior to CD at identifying deviations due to noise and occlusions, which is often desirable [31].

An obvious difference between CD and EMD is the significantly higher time complexity, which has resulted in EMD almost exclusively being reserved for use as an evaluation metric rather than for optimisation. This has also led to the introduction of various EMD approximation algorithms [24]. Notably, the auction algorithm [2] and recently the Sliced Wasserstein Distance (SWD) have been utilised specifically as point set distance metrics [14,18]. SWD is computed by projecting the source and target onto a unit sphere on a certain direction, and measuring distance between the two projections. This is repeated over different slices, and the distance is approximated using Monte Carlo estimation. Despite their relative efficiency, these approximations must still be computed iteratively in contrast to CD.

2.3 Alternative Distance Metrics

Due to the computational complexity of EMD, various CD modifications have been suggested as loss functions. These aim to preserve the efficiency of CD while providing some of the benefits of EMD. They include Density-aware CD (DCD) [31], hyperbolic CD (HyperCD) [11], Contrastive CD (InfoCD) [12], Oriented and Directional CD (OCD/DCD) [16], weighted CD [4] and learnable CD (LCD) [5]. DCD introduces two main modifications to CD [31]. First, it introduces a first-order approximation of Taylor Expansion, in order to suppress over sensitivity to outliers due to square growth. Second, it normalises the contribution of each point to the total distance by scaling each term by the total number of correspondences of the corresponding points. Essentially, this normalises the effect of density mismatches between the clouds. Unfortunately, the latter modification is not differentiable, and the scaling factor is replaced by a constant parameter when DCD is used as an objective function.

HyperCD computes the distance between correspondences in hyperbolic space rather than Euclidean space in order to increase the weight of 'good' point matches [11]. InfoCD builds on HyperCD, and attempts to maximise the lower bound of the mutual information between the two underlying geometric surfaces [12]. The authors show that this results in a regularising effect upon CD, which prevents clustering. Note that InfoCD is essentially composed of chamfer loss and an additional regularising component. The weight of the regularising component must be manually tuned. Furthermore, InfoCD is only stable for L1 loss, resulting in slower convergence compared to the L2 loss commonly utilised in CD.

OCD/DCD take into account the vector directions of each term in CD to provide additional information for single image reconstruction tasks [16]. Weighted CD weighs up larger displacement terms [4], achieving the opposite effect of the Taylor expansion used in DCD. LCD extends the concept of weighting individual terms by training a neural network to predict the ideal weights [5]. However, this results in a black box loss function/metric, leading to additional uncertainty. In summary, most CD alternatives focus on modifications to the distance calculation between correspondences or to the weighing of loss terms during aggregation. Through these modifications, they indirectly modify the correspondences in future iterations.

Based on the above findings, we seek the following properties in an improved distance metric. It must be differentiable, computationally efficient, and have a high rate of convergence. It should be able to prevent point clustering due to convergence towards local minima. Despite addressing nearly all of these requirements, EMD, or even its approximations are not feasible due to computational cost. Comparatively, CD is faster, but can cause point clustering. Modified CD algorithms attempt to bridge the gap between CD and EMD. In particular, InfoCD attempts to address the point clustering issue, but is unstable beyond L1 loss, resulting in slower convergence, and cannot directly modify clustered correspondences. The density balancing approach of DCD comes closest to our objective of addressing the density mismatches due to clustering, but lacks differentiability.

We therefore aim to modify CD to favour clouds with uniform densities without compromising differentiability, and propose our modified CD function to address this objective.

3 Proposed Method

We seek an objective function to compare a generated cloud (X) and target cloud (Y) which will enable a model to learn an accurate representation of point cloud inputs. Specifically, our objective is to improve chamfer distance in order to increase its ability to match densities between clouds, thereby reducing clustering of points, similar to EMD. Recall the three stages of point set distance computation; correspondence search, distance computation and aggregation. Previous modifications to CD focus exclusively on modifications to distance computation or aggregation. However, the distinction between CD and EMD lies solely in the correspondence search. We therefore shift our focus to improving the correspondence search.

Specifically, we attempt to take relative local density between the two clouds into account when choosing correspondences. To this end, we propose a novel loss function, labelled Uniform Chamfer Distance (UniformCD). UniformCD in a single direction $X \rightarrow Y$ is given by;

$$D_{UniformCD} = \sum_{j} ||y_i - x_j||_2 \tag{3}$$

where;

$$i = \arg\min_{i \in N} f(i) = ||y_i - x_j||_2 \times \frac{dens(y_i, X)}{dens(y_i, Y)} \tag{4}$$

where $N =$ number of points. The density ratio represents the disparity between the density of Y in the local neighbourhood of the proposed correspondence, and the density of X around the same point. Local density of a potential correspondence is measured by the inverse of the sum of squared distances to its k nearest points (kNN). In the base case of $k = 1$ this becomes;

$$dens(y_i, X) = \frac{1}{min_p ||y_i - x_p||_2} \qquad dens(y_i, Y) = \frac{1}{min_{q \neq i} ||y_i - y_q||_2} \tag{5}$$

We limit the correspondence search in Equation 4 to the local neighbourhood of a point to preserve computational efficiency. Thus, we can simply modify the nearest neighbour search in CD to a kNN to find the local neighbourhood, which is then reused in the density calculation (Equation 5) in the opposite direction.

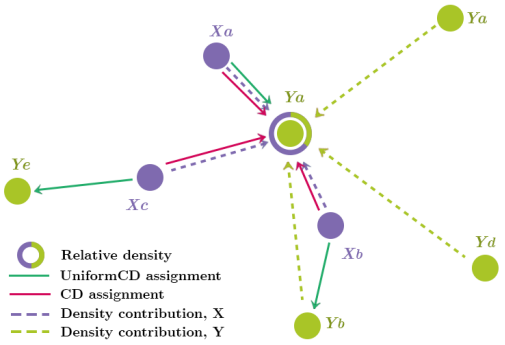

Fig. 2: Comparison between point assignment strategies of CD and UniformCD based on relative density. (Figure depicts X → Y assignment only, and demonstrates the relative density calculation for y_a, based on 3 nearest points.)

Intuitively, this loss function pushes generated points towards regions in which the density of the X is lower than the density of the Y. Consider two points x_a and x_b, whose closest point in the target is y_a (Figure 2). Under CD, both these points correspond to y_a. However under UniformCD, aside from closest distance, relative density is also used to determine point correspondence. The density of X around point y_a is high due to the presence of x_a, x_b, and x_c. However the density of Y around y_a is lower as y_b, y_c and y_d are relatively farther from y_a. Thus, the relative density of the X compared to Y is high at y_a. Comparatively, the relative density around y_b is lower, as points of X are farther from y_b. When accounting for both relative density as well as distance, x_b is likely to correspond to y_b instead of y_a. At a macro level, this prevents multiple points in the source corresponding to a single point in the target, thereby preventing clustering at local minima.

The behaviour of UniformCD under density disparities can be further examined by visualising the point-wise distances between two clouds with mismatched

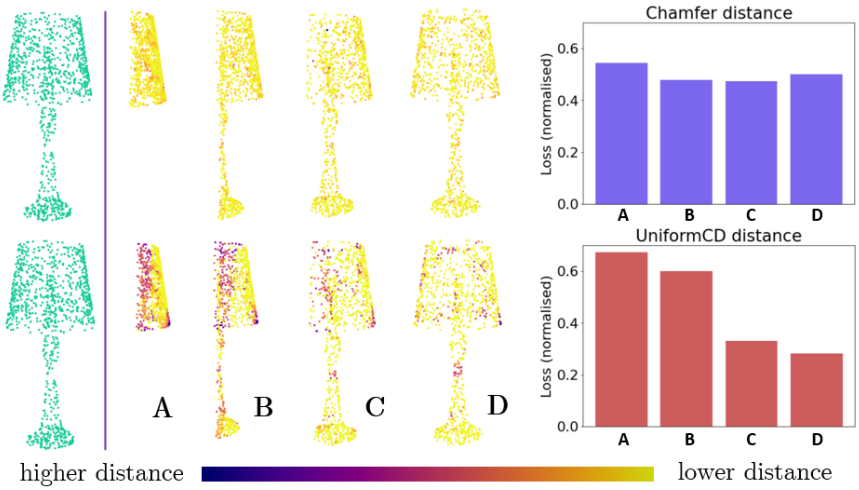

Fig. 3: Clouds A to D are generated by adding decreasing occlusions to an input cloud, followed by downsampling to *n=1048*. This results in a density disparity. The colour of each point visualises distance to its corresponding point in the target (green) based on chamfer (top) and UniformCD (bottom) distance. The total distance variation from cloud to target (right) demonstrates the sensitivity of UniformCD to density disparities.

densities (Figure 3). Notably, points in areas of mismatched density have a significantly higher point-wise loss under UniformCD in contrast to CD. This induces these points to shift towards a uniform density distribution during optimisation.

Analysis. We hypothesise that the proposed UniformCD objective function shrinks the discrepancy between CD and EMD, and moves closer towards the globally optimum correspondences of EMD. We evaluate this hypothesis in a number of ways. Consider the behaviour of various loss functions on the simplified example task of iteratively morphing a sphere into a desired shape (Figure 1). Firstly, the proposed method achieves a lower EMD score than all other CD alternatives, and outperforms CD in the chamfer distance metric (Figure 4, top). Note that alternative loss functions are able to outperform the CD objective function in the chamfer distance metric due to convergence of CD towards local minima. In both metrics, UniformCD performs closer to EMD than others.

Next, consider the relationship between X $\rightarrow$ Y point correspondences and Y $\rightarrow$ X correspondences. These are identical in the case of EMD, as it is a symmetric loss. However, these discrepancies are often dissimilar in CD, leading to imperfect alignments between the clouds. Thus the percentage of matching forward and backward correspondences can be seen as a metric for the similarity between CD and EMD, with CD converging upon EMD as this value reaches 100%. UniformCD is notably closer to EMD than other losses in this regard (Figure 4, bottom-left).

Furthermore, under EMD, each point in X corresponds to a single point in Y. However, this constraint does not hold under CD, causing multiple points

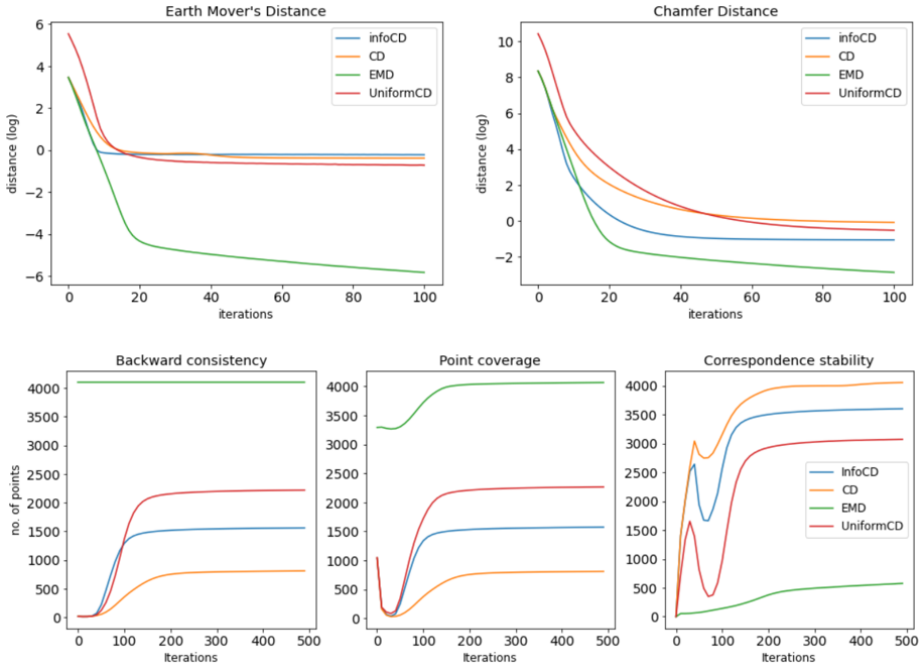

Fig. 4: Optimisation behaviour of CD, EMD, InfoCD and UniformCD losses. **Top:** CD and EMD metric variation during shape optimisation. Note that values are log normalised to highlight their disparity. UniformCD outperforms CD and InfoCD in the EMD metric, and outperforms CD under the CD metric. **Bottom-left:** Backwards consistency of correlations, as measured by the number of correspondences that are identical in both (X→ Y) and (Y→ X) directions. **Bottom-centre:** Coverage of points, as measured by the number of points in each cloud which correspond to a point in the other cloud. **Bottom-right:** Stability of point correspondences (i.e. assignments remain unchanged from one iteration to the next). $N_{points} = 4096$.

in X to correspond to a single point in Y, leaving some points in Y without a correspondence. We gauge this behaviour by measuring the change in the number of points without correspondence over time (Figure 4, bottom-centre). Similar to backward consistency, UniformCD moves closer towards the ideal behaviour of EMD. Note that the imperfect initial behaviour of EMD is caused by the EMD approximation approach used in the experiment.

Additionally, it can be observed that point correspondences stabilise as the optimisation converges (Figure 4, bottom-right). Realistically, perfect correspondences are not found during optimisation, and points converge towards a state with low loss where a small deviation of point locations causes a shift in correspondences. This is particularly evident in the case of EMD and, to a lesser degree, in UniformCD. In contrast, correspondences show high stability in chamfer loss. Note that the caveat of stability in CD is convergence towards local minima. We observe that while the volatility of correspondences reduces the stability

of UniformCD, it allows points to locate the globally optimum correspondences while only having local density information in each iteration.

Having established that UniformCD drives the behaviour of CD closer towards that of EMD, we next seek to evaluate its performance in point cloud learning and optimisation tasks.

4 Experiments

We examine the performance of UniformCD on a diverse set of datasets (MVP [19], ShapeNet [3], PCN [39], ModelNet [32] and industrial BIM elements [7]), network architectures (PointNet [21], PCN [39], VRC [19], PointAttn [8] and SeedFormer [42]) and tasks (completion, reconstruction, classification, and parametric modelling), to ensure that it is universally applicable.

Datasets

- *MVP:* This is a multi-view partial point cloud dataset rendered from partial 3D shapes captured by 26 uniformly distributed camera poses [19]. It contains 62,400 and 41,600 training and testing point sets spanning 16 categories and is used for training and evaluating completion performance.
- *ShapeNet-55*: This is a dataset consisting of 41,952 3D shape models for training and 46,765 shapes for testing, spanning 55 categories [3]. We use the subset presented in [18] with 2048 points sampled from each shape. This dataset is used for training point cloud reconstruction models.
- *PCN:* This is a subset of ShapeNet [3] spanning 8 categories [39]. It contains points rendered using 2.5D depth images from 8 view points. 2,048 points are sampled to produce the occluded cloud while the complete cloud consists of 16,384 uniformly sampled points. This dataset is used for training and evaluating completion performance.
- *ModelNet40:*. This is a 3D CAD model dataset spanning 40 categories [32], comprising 9,843 and 2,468 shapes for training and testing respectively. This dataset is used to evaluate reconstruction and classification performance.
- *Industrial BIM dataset:* This is a dataset of point cloud scans sampled from multiple camera viewpoints of 16,384 3D parametric BIM models of industrial facility elements [7]. We utilise pipe, elbow, and flange classes for testing parametric modelling performance.

Implementation Details. We use a local neighbourhood of 32 points for both correspondence search and density calculations. All models are trained with Adam optimiser [9], implemented in PyTorch and run on an RTX 3080 GPU. To compute EMD results we use an EMD approximation based on the auction algorithm introduced by Liu et al. [14] Ablation studies on k and other variables are provided in the supplementary material.

4.1 Point cloud completion

Point cloud completion is the task of inferring missing points from an occluded input. The majority of previous work on CD modifications has been evaluated ex-

clusively on point cloud completion [11,12,31]. We test our solution on MVP and PCN benchmarks, and evaluate completion performance using CD and EMD.

We compare against PCN [39], TopNet [26], MSN [14] and VRC [19], and in particular CD, InfoCD and DCD losses with VRC on the MVP benchmark (Table 1, Figure 5). We use the standard experimental setup proposed in [39], and all hyperparameters such as learning rate, point size, and number of epochs are identical to the setup in [31].

Next, we compare against FoldingNet [36], PCN, PointAttn [8] and SeedFormer [42] on the PCN benchmark (Table 2). In particular, we compare against InfoCD, HyperCD and DCD losses with SeedFormer architecture. We note that fine-tuning a CD trained model on UniformCD produce best results on this dataset. We use the experimental setup and hyperparameters used in [12] to enable comparison, and report L1 CD scores as per the norm.

For all models, UniformCD achieves significant EMD improvements in both datasets compared to models trained on CD. However, there is a negative impact on CD performance in both benchmarks, demonstrating that there is a trade-off between EMD and CD metrics. As EMD is generally considered to be more superior due to noise and occlusion sensitivity, it may be argued that biasing the model towards better EMD performance is beneficial. However, CD may also be preferable in some cases. Regardless, the results further prove that UniformCD pushes the behaviour of CD towards EMD, and that it reduces the disparity between EMD and CD. Specifically, the Pearson correlation coefficient between EMD and CD metrics for the PCN testset increases from 0.817 to 0.843 when trained on UniformCD instead of CD. Therefore, employing UniformCD as an objective function produces results that harmonise CD and EMD. Additional results are provided in the supplementary material.

Method	Metrics	Airplane	Cabinet	Car	Chair	Lamp	Sofa	Table	Watercraft	Bed	Bench	Bookshelf	Bus	Guitar	Motorbike	Pistol	Skateboard	Avg.
	CD	4.50	8.83	6.41	13.01	21.33	9.90	12.86	9.46	20.00	10.26	14.63	4.94	1.73	6.17	5.84	5.76	9.78
PCN	EMD	4.70	7.99	5.75	6.90	11.99	5.32	6.60	5.40	9.84	4.85	7.87	5.24	10.56	4.93	4.86	5.59	6.80
	CD	4.05	10.61	7.15	14.22	18.86	10.75	15.85	9.70	21.64	10.45	14.91	5.76	2.24	6.53	6.07	5.78	10.84
PCN-ours	EMD	2.89	3.53	3.10	4.65	7.79	3.51	4.53	4.02	5.49	3.63	4.26	2.58	3.03	3.46	3.68	2.21	4.07
	CD	4.06	9.08	6.64	13.11	19.25	9.78	14.36	9.66	22.33	9.73	15.51	5.13	1.86	6.25	5.81	4.99	10.29
PCN++	EMD	3.44	3.75	3.15	4.65	8.00	3.56	4.69	4.22	6.13	3.85	4.39	2.62	2.78	3.60	3.71	3.07	4.27
	CD	4.12	9.84	7.44	13.26	18.64	10.77	12.95	8.98	19.99	9.21	16.06	5.47	2.36	7.06	7.04	4.68	10.30
TopNet	EMD	4.89	6.30	4.07	7.01	10.75	6.47	7.50	4.68	8.09	6.27	6.80	3.50	4.21	4.26	6.02	3.49	6.18
	CD	2.73	8.92	6.50	10.75	13.37	9.26	10.17	7.70	17.27	6.64	12.1	5.21	1.37	4.59	4.62	3.38	7.99
MSN	EMD	2.75	4.02	3.47	4.44	6.28	3.74	4.46	3.82	5.27	3.34	4.28	2.92	2.07	3.30	3.62	2.21	3.94
	CD	2.20	7.92	5.60	7.49	8.15	7.45	7.52	5.20	11.90	4.88	7.39	4.53	1.15	3.90	3.44	3.22	6.09
VRC	EMD	3.03	7.57	6.14	5.49	6.15	5.80	4.65	4.97	6.58	3.45	5.28	6.59	3.08	4.45	4.56	3.20	5.27
	CD	2.72	9.03	6.58	9.93	11.53	9.38	9.80	6.71	17.22	6.88	10.34	5.32	1.39	4.47	4.62	4.69	7.87
VRC-EMD	EMD	2.50	3.65	3.23	4.15	5.31	3.61	3.93	3.58	5.17	3.19	3.97	2.69	2.08	3.06	3.48	2.29	3.62
	CD	2.22	8.00	**5.41**	7.88	8.28	7.94	8.89	5.46	14.76	5.78	9.37	4.44	1.30	**3.59**	3.43	**2.39**	6.51
VRC-DCD	EMD	2.29	4.43	3.46	3.92	4.98	3.98	3.89	3.51	5.34	3.13	3.91	3.29	2.21	3.02	3.38	2.39	3.67
	CD	**2.03**	**7.88**	**5.41**	**7.31**	**7.92**	**7.22**	**7.3**	**5.01**	**11.67**	**4.65**	**7.14**	**4.30**	**0.97**	4.68	**3.19**	3.04	5.87
VRC-InfoCD	EMD	2.68	7.26	5.83	5.15	5.82	5.49	4.36	4.68	6.22	3.13	4.97	6.26	2.77	4.13	4.15	2.89	4.97
	CD	2.56	10.36	6.68	9.58	11.83	9.92	11.57	6.49	17.94	7.15	12.31	5.79	1.52	4.10	4.13	3.80	8.25
VRC-ours	EMD	**2.07**	**3.41**	**2.87**	**3.52**	**3.93**	**3.44**	**3.59**	**2.92**	**4.53**	**2.83**	**3.51**	**2.57**	**1.48**	**2.44**	**2.42**	**1.78**	**3.09**

Table 1: Completion results on MVP dataset in terms of CD ($\times 10^4$) and EMD ($\times 10^2$). Objective function is CD unless otherwise specified. Lower is better.

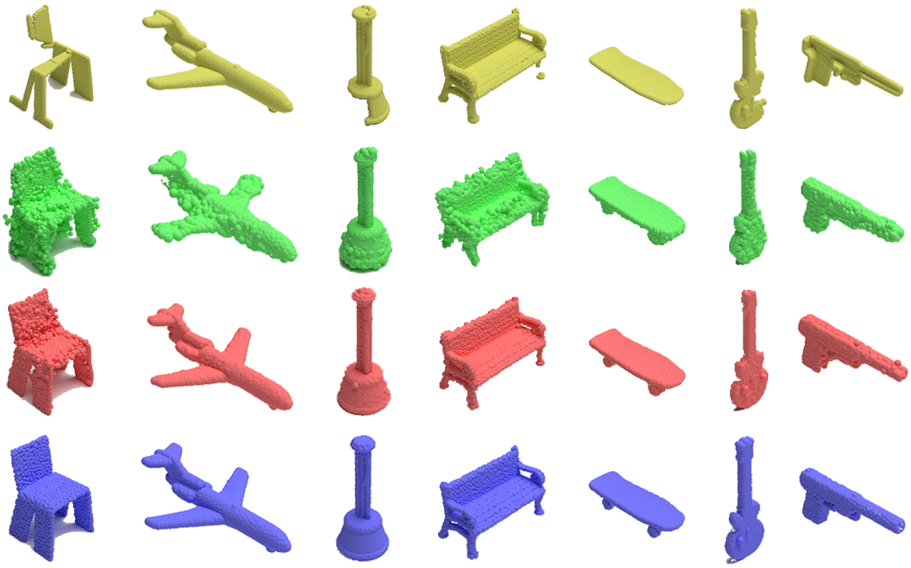

Fig. 5: Completion results on MVP dataset. From top to Bottom: Partial input (yellow), VRC-CD (green), VRC-UniformCD (red), ground truth (blue).

4.2 Point cloud reconstruction

We test reconstruction performance using the PointNet based autoencoder architecture and experimental setup proposed in [18] on CD and EMD metrics. We compare results against CD, EMD, SWD [18] and InfoCD (Table 3a). As was the case with point cloud completion, UniformCD improves EMD scores, at a slight cost to CD scores. We observe that training time per iteration for UniformCD increases by around 18% compared to CD due to the increased time cost of the neighbour search. This can be adjusted by lowering k (Table 4b).

4.3 Self-supervised learning

One approach to bypass the hurdle of identifying the optimal metric for evaluating a model trained on a distance metric is to directly measure model performance on a downstream task. This essentially measures the effectiveness of the 3D representation learned by the model. This can be evaluated by using the autoencoder trained above for a downstream task on an unseen dataset. Specifically, we train a classifier on ModelNet using vector embeddings generated from the above network. We follow the same experimental setup as [18], utilising a 256-dimension vector for the embedding, and an SGD optimiser. Classification accuracy is compared against CD, EMD, SWD and InfoCD. (Table 3b). UniformCD achieves the highest classification accuracy among all loss functions. This demonstrates that achieving a balance between EMD and CD metrics is beneficial for downstream tasks.

Method	Metric	Plane	Cabinet	Car	Chair	Lamp	Couch	Table	Boat	Avg.
	CD	9.49	15.80	12.61	15.55	16.41	15.97	13.65	14.99	14.31
FoldingNet	EMD	15.64	22.13	17.45	29.74	32.00	24.56	19.00	21.88	22.08
	CD	5.50	22.70	10.63	8.70	11.00	11.34	11.68	8.59	11.27
PCN	EMD	4.66	9.77	5.72	8.68	16.95	8.34	8.08	7.51	8.71
	CD	3.87	9.00	**7.63**	7.43	5.90	8.68	6.32	6.09	6.86
PointAttN	EMD	1.15	2.51	2.31	1.91	**1.46**	2.85	1.48	1.79	1.93
	CD	3.85	9.05	8.06	7.06	5.21	8.85	6.05	5.85	6.74
SeedFormer	EMD	1.81	4.09	3.60	2.75	2.20	4.12	2.24	2.66	2.93
	CD	16.42	26.23	21.08	20.06	18.30	26.51	18.23	18.22	24.52
SeedFormer -DCD	EMD	13.05	18.55	12.90	14.58	9.82	19.47	16.22	14.20	16.84
	CD	3.72	**8.71**	7.79	6.83	5.11	**8.61**	**5.82**	5.76	**6.54**
SeedFormer -HyperCD	EMD	1.63	3.62	3.16	2.57	2.06	3.78	1.99	2.40	2.65
	CD	**3.70**	8.79	7.72	**6.82**	**5.08**	8.63	5.84	**5.70**	**6.54**
SeedFormer -infoCD	EMD	1.75	3.94	3.43	2.72	2.16	4.04	2.09	2.58	2.84
	CD	5.09	10.91	9.05	9.87	8.05	12.01	8.44	7.87	8.91
PointAttN -Ours	EMD	1.06	**1.98**	1.72	1.85	1.62	2.34	1.56	1.56	1.71
	CD	4.80	10.53	8.91	9.31	7.38	11.33	7.91	7.29	8.45
PointAttN -Ours(f/tune)	EMD	**1.00**	**1.89**	**1.67**	**1.72**	**1.46**	**2.20**	**1.47**	**1.49**	**1.61**
	CD	5.94	12.30	10.30	10.37	9.33	13.35	8.84	8.49	9.86
SeedFormer -Ours	EMD	1.80	3.08	2.62	2.62	2.39	3.62	2.15	2.35	2.57
	CD	4.61	10.32	9.19	8.31	6.28	11.23	7.04	6.91	7.99
SeedFormer -Ours(f/tune)	EMD	1.54	2.58	2.43	2.10	1.78	2.98	1.78	1.94	2.14

Table 2: Completion results on PCN dataset in terms of L1 CD ($\times 10^4$) and EMD ($\times 10^2$). Objective function is CD unless otherwise specified. Lower is better.

	CD		EMD	
Objective fn.	mean	stddev	mean	stddev
CD	8.541	5.883	5.88	2.0
SWD (SSW)	**7.403**	4.956	8.31	2.6
InfoCD	13.52	3.32	9.44	0.037
UniformCD	9.145	6.44	**5.58**	2.0

(a) Point cloud reconstruction results in terms of CD ($\times 10^4$) and EMD ($\times 10^2$). PointNet based autoencoder trained on ShapeNetCore55. Lower is Better.

Objective fn.	Accuracy (%)
CD	83.9
EMD	84.4
SWD (SSW)	86.8
InfoCD	86.3
UniformCD	**87.12**

(b) Classification results using self-supervised embeddings from encoder trained on ShapeNetCore55. Higher is Better.

Table 3: Reconstruction and Classification results on ModelNet40.

	elbow		pipe		flange	
objective fn.	CD	EMD	CD	EMD	CD	EMD
CD	12.08	2.75	2.27	0.91	16.43	6.13
EMD	43.13	1.01	9.38	0.26	51.15	1.47
InfoCD	12.04	2.85	2.36	0.89	17.32	6.28
UniformCD	10.87	2.37	2.83	0.74	21.14	4.87

(a) Parametric modelling results on industrial facility elements in terms of CD ($\times 10^4$) and EMD ($\times 10^2$). Lower is better.

Objective fn.	time (ms)
CD	143
EMD (auction alg.)	757
InfoCD	144
SWD	153
UniformCD	169

(b) Comparison of training time per iteration on PointNet autoencoder.

Table 4: Parametric modelling performance and autoencoder run times.

4.4 Parametric modelling

Parametric modelling involves inferring the underlying geometric parameters of points sampled from a parametric 3D shape. Minimising 3D loss between a predicted shape and the ground truth point has recently been utilised as a method of recovering geometric parameters from point clusters [7]. This task is notably different from the previous tasks, as the geometric parameters of a given shape are directly optimised, instead of optimising the weights of a neural network which predicts points based on an input. We use the experimental setup proposed in [7] including all hyperparameters, and replace the loss function for the parameter optimisation step with our proposed loss.

UniformCD outperforms all CD variants on EMD, and achieves the best CD values on elbow elements (Table 4a). However, directly optimising EMD achieves a significantly better EMD value than all CD variants. It is worth noting that while the EMD objective function substantially outperforms all other alternatives on direct optimisation tasks, this improvement is much lower when EMD is used for model training. This can clearly be seen in VRC model performance on point cloud completion (Table 1).

5 Conclusion

We present UniformCD, a novel point set objective function for 3D representation learning and iterative optimisation tasks. Distinct from previous CD modifications, we utilise an improved correspondence search to encourage uniform density distributions during optimisation. This prevents clustering at convergence and produces outputs that match the local densities of the input clouds. The proposed loss is agnostic to network architecture and pushes the dynamic behaviour of CD significantly closer towards EMD. This results in improved performance on a variety of tasks such as point cloud completion, self-supervised learning and 3D model parametric optimisation.

Limitations. The method reduces the smoothness of correspondences compared to CD, particularly at convergence. Analysing the effect of the variability of correspondences at convergence on model performance would be a useful avenue for future exploration. Additionally, due to the larger search radius of kNN search, UniformCD is slightly (∼18%) slower that vanilla CD.

Acknowledgements. This research has received funding from EPSRC, AVEVA Group plc and British Petroleum plc.

References

1. Achlioptas, P., Diamanti, O., Mitliagkas, I., Guibas, L.: Learning Representations and Generative Models for 3D Point Clouds (7 2017), http://arxiv.org/abs/1707.02392
2. Bertsekas, D.P.: The Auction Algorithm for Assignment and Other Network Flow Problems: A Tutorial. Interfaces **20**(4), 133–149 (8 1990). https://doi.org/10.1287/inte.20.4.133

3. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., Yu, F.: ShapeNet: An Information-Rich 3D Model Repository. Tech. Rep. arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago (2015)
4. Furuya, T., Liu, W., Ohbuchi, R., Kuang, Z.: Hyperplane patch mixing-and-folding decoder and weighted chamfer distance loss for 3D point set reconstruction. Visual Computer **39**(10), 5167–5184 (10 2023). https://doi.org/10.1007/s00371-022-02652-6
5. Huang, T., Liu, Q., Zhao, X., Chen, J., Liu, Y.: Learnable Chamfer Distance for point cloud reconstruction. Pattern Recognition Letters **178**, 43–48 (2 2024). https://doi.org/10.1016/j.patrec.2023.12.015, https://linkinghub.elsevier.com/retrieve/pii/S016786552300363X
6. Imperoli, M., Pretto, A.: D CO: Fast and Robust Registration of 3D Textureless Objects Using the Directional Chamfer Distance. In: International conference on computer vision systems. pp. 316–328 (2015)
7. Jayasinghe, H., Brilakis, I.: Learnable Geometry and Connectivity Modelling of BIM Objects. In: 34th British Machine Vision Conference 2023, BMVC 2023, Aberdeen, UK, November 20-24, 2023. BMVA (2023), https://papers.bmvc2023.org0305.pdf
8. Jun Wang Ying Cui, D.G.J.L.Q.L.C.S.: PointAttN: You Only Need Attention for Point Cloud Completion. In: Association for the Advancement of Artificial Intelligence (AAAI) (2 2024)
9. Kingma, D.P., Ba, J.: Adam: A Method for Stochastic Optimization. arXiv preprint (12 2014), http://arxiv.org/abs/1412.6980
10. Li, S., Liu, M., Walder, C.: EditVAE: Unsupervised parts-aware controllable 3D point cloud shape generation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 1386–1394 (2022)
11. Lin, F., Yue, Y., Hou, S., Yu, X., Xu, Y., Yamada, K.D., Zhang, Z.: Hyperbolic Chamfer Distance for Point Cloud Completion. Tech. rep., https://github.com/Zhang-VISLab.
12. Lin, F., Yue, Y., Zhang, Z., Hou, S., Yamada, K.D., Kolachalama, V.B., Saligrama, V.: InfoCD: A Contrastive Chamfer Distance Loss for Point Cloud Completion. In: Neurips (2023), https://github.com/Zhang-VISLab/NeurIPS2023-InfoCD.
13. Liu, M.Y., Tuzel, O., Veeraraghavan, A., Chellappa, R.: Fast directional chamfer matching. In: 2010 IEEE Computer Society conference on computer vision and pattern recognition. pp. 1696–1703 (2010)
14. Liu, M., Sheng, L., Yang, S., Shao, J., Hu, S.M.: Morphing and Sampling Network for Dense Point Cloud Completion. Proceedings of the AAAI Conference on Artificial Intelligence **34**(07), 11596–11603 (4 2020). https://doi.org/10.1609/aaai.v34i07.6827
15. Liu, X., Liu, X., Liu, Y.S., Han, Z.: Spu-net: Self-supervised point cloud upsampling by coarse-to-fine reconstruction with self-projection optimization. IEEE Transactions on Image Processing **31**, 4213–4226 (2022)
16. Lu, J., Li, Z., Bai, J., Yu, Q.: Oriented and Directional Chamfer Distance Losses for 3D Object Reconstruction From a Single Image. IEEE Access **10**, 61631–61638 (2022). https://doi.org/10.1109/ACCESS.2022.3179109
17. Milani, S.: Adae: Adversarial distributed source autoencoder for point cloud compression. In: 2021 IEEE International Conference on Image Processing (ICIP). pp. 3078–3082 (2021)

18. Nguyen, T., Pham, Q.H., Le, T., Pham, T., Ho, N., Hua, B.S.: Point-set Distances for Learning Representations of 3D Point Clouds. In: 2021 IEEE International Conference on Computer Vision (ICCV) (2021)
19. Pan, L., Chen, X., Cai, Z., Zhang, J., Zhao, H., Yi, S., Liu, Z.: Variational Relational Point Completion Network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8524–8533 (2021)
20. Poppinga, J., Pfingsthorn, M., Schwertfeger, S., Pathak, K., Birk, A.: Optimized Octtree Datastructure and Access Methods for 3D Mapping. In: 2007 IEEE International Workshop on Safety, Security and Rescue Robotics. pp. 1–6. IEEE (9 2007). https://doi.org/10.1109/SSRR.2007.4381275
21. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition (12 2017), http://arxiv.org/abs/1612.00593
22. Rakotosaona, M.J., La Barbera, V., Guerrero, P., Mitra, N.J., Ovsjanikov, M.: PointCleanNet: Learning to Denoise and Remove Outliers from Dense Point Clouds (1 2019), http://arxiv.org/abs/1901.01060
23. Sharma, G., Liu, D., Maji, S., Kalogerakis, E., Chaudhuri, S., Měch, R.: ParSeNet: A Parametric Surface Fitting Network for 3D Point Clouds. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). vol. 12352 LNCS, pp. 261–276. Springer Science and Business Media Deutschland GmbH (2020). https://doi.org/10.1007/978-3-030-58571-6{_}16
24. Shirdhonkar, S., Jacobs, D.W.: Approximate earth mover's distance in linear time. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8. IEEE (6 2008). https://doi.org/10.1109/CVPR.2008.4587662, http://ieeexplore.ieee.org/document/4587662/
25. Sun, Y., Wang, Y., Liu, Z., Siegel, J., Sarma, S.: Pointgrow: Autoregressively learned point cloud generation with self-attention. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 61–70 (2020)
26. Tchapmi, L.P., Kosaraju, V., Rezatofighi, H., Reid, I., Savarese, S.: TopNet: Structural Point Cloud Decoder. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 383–392 (2019). https://doi.org/10.1109/CVPR.2019.00047
27. Tian, X., Ran, H., Wang, Y., Zhao, H.: GeoMAE: Masked Geometric Target Prediction for Self-supervised Point Cloud Pre-Training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023), https://github.com/Tsinghua-
28. Wang, H., Liu, Q., Yue, X., Lasenby, J., Kusner, M.J.: Unsupervised Point Cloud Pre-training via Occlusion Completion. In: Proceedings of the IEEE/CVF international conference on computer vision (2018), https://github.com/hansen7/OcCo
29. Wang, J., Zhu, H., Ma, Z., Chen, T., Liu, H., Shen, Q.: Learned point cloud geometry compression. arXiv preprint arXiv:1909.12037 (2019)
30. Wen, X., Xiang, P., Han, Z., Cao, Y.P., Wan, P., Zheng, W., Liu, Y.S.: Pmp-net: Point cloud completion by learning multi-step point moving paths. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7443–7452 (2021)
31. Wu, T., Pan, L., Zhang, J., WANG, T., Liu, Z., Lin, D.: Balanced Chamfer Distance as a Comprehensive Metric for Point Cloud Completion. In: Ranzato, M.,

Beygelzimer, A., Dauphin, Y., Liang, P.S., Vaughan, J.W. (eds.) Advances in Neural Information Processing Systems. vol. 34, pp. 29088–29100. Curran Associates, Inc. (2021), https://proceedings.neurips.cc/paper_files/paper/2021/file/f3bd5ad57c8389a8a1a541a76be463bf-Paper.pdf

32. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3D ShapeNets: A deep representation for volumetric shapes. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1912–1920 (2015). https://doi.org/10.1109/CVPR.2015.7298801
33. Xiang, P., Wen, X., Liu, Y.S., Cao, Y.P., Wan, P., Zheng, W., Han, Z.: Snowflake Point Deconvolution for Point Cloud Completion and Generation with Skip-Transformer. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–18 (2 2022). https://doi.org/10.1109/TPAMI.2022.3217161, https://ieeexplore.ieee.org/document/9928787/
34. Xie, H., Yao, H., Zhou, S., Mao, J., Zhang, S., Sun, W.: GRNet: Gridding Residual Network for Dense Point Cloud Completion. In: European Conference on Computer Vision. pp. 365–381 (6 2020). https://doi.org/10.1007/978-3-030-58545-7{_}21, https://link.springer.com/10.1007/978-3-030-58545-7_21
35. Yang, G., Huang, X., Hao, Z., Liu, M.Y., Belongie, S., Hariharan, B.: Pointflow: 3d point cloud generation with continuous normalizing flows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4541–4550 (2019)
36. Yang, Y., Feng, C., Shen, Y., Tian, D.: FoldingNet: Point Cloud Auto-encoder via Deep Grid Deformation. Tech. rep., http://www.merl.com/research/
37. Yu, X., Rao, Y., Wang, Z., Liu, Z., Lu, J., Zhou, J.: PoinTr Diverse Point Cloud Completion With Geometry-Aware Transformers. In: Proceedings of the IEEECVF International Conference on Computer Vision (ICCV). pp. 12498–12507 (10 2021)
38. Yu, X., Tang, L., Rao, Y., Huang, T., Zhou, J., Lu, J.: Point-BERT: Pre-training 3D Point Cloud Transformers with Masked Point Modeling. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 19291–19300. IEEE (6 2022). https://doi.org/10.1109/CVPR52688.2022.01871, https://ieeexplore.ieee.org/document/9880161/
39. Yuan, W., Khot, T., Held, D., Mertz, C., Hebert, M.: PCN: Point Completion Network. In: 2018 International Conference on 3D Vision (3DV). pp. 728–737. IEEE (9 2018). https://doi.org/10.1109/3DV.2018.00088, https://ieeexplore.ieee.org/document/8491026/
40. Zamorski, M., Zieba, M., Klukowski, P., Nowak, R., Kurach, K., Stokowiec, W., Trzciński, T.: Adversarial autoencoders for compact representations of 3D point clouds. Computer Vision and Image Understanding **193**, 102921 (2020)
41. Zhang, C., Shi, J., Deng, X., Wu, Z.: Upsampling autoencoder for self-supervised point cloud learning. arXiv preprint arXiv:2203.10768 (2022)
42. Zhou, H., Cao, Y., Chu, W., Zhu, J., Lu, T., Tai, Y., Wang, C.: SeedFormer: Patch Seeds Based Point Cloud Completion with Upsample Transformer. In: ECCV 2022: 17th European Conference. pp. 416–432 (2022). https://doi.org/10.1007/978-3-031-20062-5{_}24