Shi, L., Zhang, G., Cao, Q. , Zhang, L., Cen, Y. and Cen, Y. (2024) DCPoint: global-local dual contrast for self-supervised representation learning of 3D point clouds. *IEEE Sensors Journal*, (doi: 10.1109/JSEN.2024.3405079)

Copyright © 2024 IEEE. Reproduced under a Creative Commons Attribution 4.0 International License.

For the purpose of open access, the author(s) has applied a Creative Commons Attribution license to any Accepted Manuscript version arising.

https://eprints.gla.ac.uk/327579/

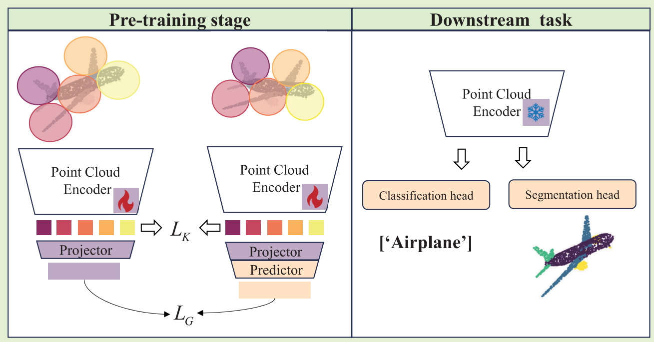
Deposited on: 05 June 2024

Enlighten – Research publications by members of the University of Glasgow
https://eprints.gla.ac.uk

DCPoint: Global–Local Dual Contrast for Self-Supervised Representation Learning of 3-D Point Clouds

Lu Shi, Guoqing Zhang, Qi Cao, Linna Zhang, Yigang Cen, and Yi Cen

***Abstract*—In recent years, 3-D vision has gained increasing prominence in practical applications such as autonomous driving and robotics. However, the scarcity of large labeled point cloud datasets continues to be a bottleneck for deep networks. Self-supervised representation learning (SRL) has emerged as an effective approach to alleviate this issue by pretraining general feature encoders without requiring human annotations. Existing contrastive SRL methods for 3-D point clouds have predominantly concentrated on object representations from a global or point perspective. They overlook essential local geometry information, thereby constraining the generalizability of pretrained models. To address these challenges, we propose a local contrast module as an intermediate level between the scene and point levels. It is then integrated with a global contrast module to form a dual contrast method known as DCPoint. The local contrast module operates on pointwise representations of objects and designs contrastive pairs based on the spatial information of point clouds. It effectively addresses the challenges posed by the sparsity and irregularity of point clouds and imperfect partition issues. The pointwise local contrast module aims to enhance the internal connections between the components within the point cloud, while the global contrast module introduces semantic information about individual instances. Experimental results demonstrate the effectiveness of DCPoint across various downstream tasks on synthetic and real-world datasets. It consistently outperforms previously reported SRL methods and the randomly initialized counterparts. In addition, the proposed local contrast module can enhance the performances of other SRL methods. Our source codes of this research are available at https://github.com/UnderTheMangoTree/DCPoint.git.**



***Index Terms*—3-D point clouds, contrastive learning, deep learning, self-supervised representation learning (SRL).**

I. Introduction

THREE-DIMENSIONAL vision tasks are fundamental perception tasks for machines to understand the physical world like a human. Therefore, 3-D scene understanding methods have been widely applied in various practical applications, including robotics [1], autonomous driving [2], and human–robot interaction [3]. Point clouds, as an essential format of 3-D data, preserve the original geometric information of objects in 3-D space. With the advent of powerful deep learning methods, promising results have been reported in using point clouds for various 3-D tasks [4], [5], [6], [7]. However, training complex deep learning models requires large-scale human-annotated training data. It is laborious and time-consuming due to the inherent ambiguity of 3-D views and the subjectivity of human perception [8].

In this article, we investigate self-supervised representation learning (SRL) to mitigate the 3-D point cloud annotation challenges. SRL pretrains models with unlabeled data to extract general representations of objects. These learned representations can be transferred to various downstream tasks by fine-tuning the pretrained models with fewer labeled data. Many works in the 2-D domain have demonstrated the feasibility of SRL [9], [10], [11]. In recent years, SRL of 3-D point clouds has attracted increasing attention [12], [13], [14], [15].

Manuscript received 11 March 2024; revised 15 May 2024; accepted 21 May 2024. This work was supported in part by Beijing Natural Science Foundation under Grant L231012 and in part by the National Natural Science Foundation of China under Grant 62062021. The associate editor coordinating the review of this article and approving it for publication was Dr. Zhenghua Chen. *(Corresponding author: Yigang Cen.)*

Lu Shi, Guoqing Zhang, and Yigang Cen are with the Institute of Information Science and Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing Jiaotong University, Beijing 100044, China (e-mail: lu_shi@bjtu.edu.cn; gq.zhang@bjtu.edu.cn; ygcen@bjtu.edu.cn).

Qi Cao is with the School of Computing Science, University of Glasgow, Singapore 567739 (e-mail: Qi.Cao@glasgow.ac.uk).

Linna Zhang is with the School of Mechanical Engineering, Guizhou University, Guiyang 550025, China (e-mail: zln770808@163.com).

Yi Cen is with the School of Information Engineering, Minzu University of China, Beijing 100081, China (e-mail: yi_cen@126.com).

Digital Object Identifier 10.1109/JSEN.2024.3405079

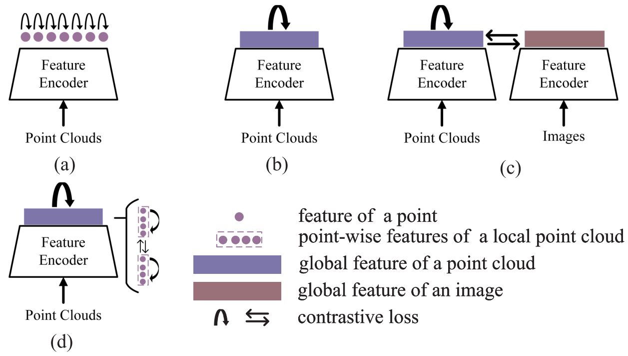


Fig. 1. Contrastive SRL methods of 3-D point clouds. (a) PointContrast [19], (b) STRL [20], (c) CrossPoint [16], and (d) our proposed DCPoint, which is different from other methods. It simultaneously considers the internal structural information and the latent classical consistency by the global–local dual contrast.

Contrastive SRL, hereinafter referred to as contrastive SRL, has demonstrated remarkable performances in 2-D and 3-D domains [16], [17]. It focuses on the similarity between different objects in the representation space [18]. A critical distinction among contrastive SRL methods lies in the attention scope and information granularity of the representation space. The existing contrastive SRL methods for 3-D point clouds predominantly concentrate on contrasting global scene representations or point representations of objects. A few examples are reported in the literature such as PointContrast [19], STRL [20], and CrossPoint [16], as shown in Fig. 1(a)–(c). However, an exclusive emphasis on global-level representation overlooks detailed information about objects. It focuses solely on point-level representation, which may disregard instance-level characteristics. These mono-perspective contrastive SRL methods will be further discussed in Section II. To address these issues, we propose an intermediate level of contrast, termed local contrast. Next we incorporate it with global contrast to form a global–local dual contrast method, as shown in Fig. 1(d). The proposed dual contrast method, denoted as DCPoint, fills the absence of multiperspective contrastive SRL methods for 3-D point clouds.

The local contrast module aims to capture the correlations between the components of objects. It is impractical to directly apply 2-D local contrast techniques to construct 3-D local contrastive sample pairs due to the sparsity, irregular spatial distribution, and permutation invariance inherent in 3-D point clouds. To address this challenge, previous 3-D SRL methods introduce the proposal extractor and self-similarity model [12], [21], at the cost of increasing the computational load. In this article, we propose a pointwise local contrast module, which defines local contrastive sample pairs through spherical partition in the Euclidean space of point clouds. To enhance interpartition consistency and intrapartition discrimination of objects, this module shrinks the representation distances between a center and its neighbors within the same partition. While it increases the representation distances between the centers of different local regions. Compared with previous local SRL methods, our pointwise local contrast module adapts to the unique properties of point clouds. It mitigates the imperfect local partition problem arising from the absence of the ground truth [22]. For instance, a randomly divided local point cloud of a plane may include points from both the fuselage and the wings.

Our global contrast module proves beneficial in learning data invariance. Considering its stability, we use an asymmetric architecture to shrink the global representation distances between two augmented views of a point cloud. Significantly, our global contrast module is streamlined by learning exclusively from semantic-related pairs, drawing inspiration from BYOL [23].

By incorporating the intermediate level of contrastive learning with the global scene level, our DCPoint overcomes the limitations of mono-perspective SRL methods. It boosts the discriminative power of the learned representations.

We evaluate DCPoint across three downstream tasks to illustrate its effectiveness: 3-D object classification, part segmentation, and semantic segmentation. Two datasets are used in the classification evaluation: the synthetic dataset ModelNet40 [24] and the real-world dataset ScanObjectNN [25]. It is observed that DCPoint consistently outperforms the state-of-the-art SRL methods in linear classification accuracy. Specifically, DCPoint achieves an accuracy of 91.5% on ModelNet40 and 82.3% on ScanObjectNN. Moreover, DCPoint surpasses its randomly initialized counterparts and other SRL methods, in the evaluations with fine-tuning and few-shot learning (FSL). Furthermore, compared with its closest competitor, STRL [20], and randomly initialized counterparts, DCPoint demonstrates notable advancements in the part segmentation dataset ShapeNetPart [26] and the semantic segmentation dataset S3DIS [27]. Particularly in the context of semi-supervised learning, DCPoint exhibits promising improvements. To gain further insights into the effectiveness of DCPoint, we conduct abundant ablation studies to examine the componentwise contributions of our global and local contrast modules. The results confirm the significance of both the components in enhancing the overall performance of DCPoint. In addition, our experiments reveal that the proposed local contrast module can effectively improve the performances of other SRL methods [20], [28], which implies its potential as a valuable enhancement to the existing approaches.

The main contributions of this research are summarized as follows:

1) We introduce a local contrast module for 3-D point clouds to capture crucial structural information of objects. It improves the consistency and discrimination of various local regions on the representation space. This module constructs contrastive sample pairs based on the spatial heuristic of 3-D point clouds. It effectively addresses the local partition problem arising from the absence of ground truth and accommodates the unique properties inherent in point clouds.
2) We introduce DCPoint, a dual contrast method that integrates our local contrast module with a global contrast module. DCPoint captures information at multiple levels of granularity and perspectives. It enables a more comprehensive and nuanced understanding of 3-D point clouds.

3) We evaluate DCPoint across various downstream tasks on four widely used synthetic and real-world datasets, where our DCPoint outperforms its randomly initialized counterparts and other SRL methods. The proposed local contrast module can further enhance the generalization capabilities of other SRL methods.

II. Related Work

With the advancement of deep learning techniques, the scale and quality of training data gradually become a bottleneck [18]. Labeling a large dataset is time-consuming and labor-intensive. Therefore, unsupervised learning becomes popular in the research area of artificial intelligence, which aims to train neural networks without human annotations [29]. As the intermediate product of unsupervised learning, SRL has gained considerable attention and demonstrated remarkable efficacy in 2-D vision tasks [17], [23]. Researchers have recently explored the SRL methods of 3-D point clouds, which mainly comprise context-based and generative methods [8].

A. Context-Based SRL of 3-D Point Clouds

Context-based SRL of 3-D point clouds intends to learn the different contexts of point clouds, encompassing contrastive and structural SRL.

Contrastive SRL of 3-D point clouds is one of the mainstream SRL types. It aims to capture the potential semantics from constructed positive and negative pairs [30]. Drawing inspiration from the success of contrastive SRL in 2-D vision tasks, numerous researchers have explored the effectiveness of such techniques in 3-D vision tasks [31], [32], [33]. For example, PointContrast [19] extends MoCo [17] to the point-level contrast, where a positive pair comprises two points of two views generated from a point cloud. STRL [20] adopts the framework of BYOL [23] to learn the representations of 3-D point clouds. CrossPoint [16] and Simipu [34] introduce cross-modal contrastive SRL methods by incorporating 3-D–2-D consistency in addition to 3-D self-consistency. Different from the above mono-perspective methods, our DCPoint simultaneously uses global and local contrast to capture the semantic and geometric representations of objects.

Structural SRL of 3-D point clouds aims to capture geometric information of point clouds by predicting their spatial information. It provides accurate geometric representation and natural geometric labels. For example, self-orientation [28] pretrains a model to predict the rotation angle of objects. It uses orientation information as a supervision signal without relying on human annotations. However, the disparity between the classification-related information and the one-sided geometric information limits the generality of structural SRL methods. Therefore, the recent work [35] uses structural SRL as the auxiliary pretext task. Differently, our local contrast module captures the latent structural information by distinguishing between local positive and negative sample pairs. It can be as a plug-and-play module, which further enhances the generality of structural SRL methods.

B. Generative SRL of 3-D Point Clouds

Generative SRL of 3-D point clouds aims to generate original and complete point clouds from their destroyed counterparts. Through the reconstruction process, the point cloud encoder can capture the association between local and global areas. For instance, Jigsaw [36] uses randomly disrupted 3-D point clouds as the input and aims to generate the original version. OcCo [37] first masks a portion of point clouds from specific camera views and then reconstructs the complete point clouds from the masked version. Point-MAE [14] reconstructs the masked content of a point cloud by masked autoencoding with transformer [38]. ACT [39] is reported to capture the latent knowledge of 3-D point clouds from natural language and 2-D vision with cross-modal reconstruction task.

III. Method

In this section, we elaborate on the proposed global–local dual contrast SRL method: DCPoint. We start with the preliminaries in Section III-A, including the problem formulation and notations of contrastive SRL. Then, we briefly describe our SRL method DCPoint in Section III-B. Next, the crucial components, i.e., local contrast (see Section III-C), global contrast (see Section III-D), and the global–local joint objective (see Section III-E), are described in detail.

A. Preliminaries

Due to the tedious and time-consuming nature of labeling point clouds, the number of large-scale annotated datasets remain limited in the field of 3-D computer vision tasks [8]. In this article, we aim to alleviate the dependence of deep networks on human annotations in the 3-D point cloud domain through SRL. SRL guides models to extract object-specific features through pretext tasks that do not require human annotations, e.g., reconstruction and contrastive tasks. It serves as a beneficial initialization for the feature encoder because it imparts the model with an understanding of object features and their relationships. It can significantly enhance the model performance on downstream tasks. SRL equips the model with a more robust and generalized representation of objects in the pretraining process. As such, the models will not easily overfit with few labeled training data compared with the random initialization [29].

As an essential branch of SRL, contrastive SRL has demonstrated superior performances in the 2-D and 3-D domains. Two critical issues of contrastive SRL are *positive pairs* and *negative pairs*. Contrastive SRL aims to reduce the embedding distances between positive pairs and enlarge the embedding distances between negative pairs. InfoNCE loss [17] is a widely used training objective function, which is defined as follows:

$$L_{\text{info}} = -\log \frac{\exp\left(f_q(x)^T \cdot f_k(x^+)/\tau\right)}{\sum_k \exp\left(f_q(x)^T \cdot f_k(x^k)/\tau\right)} \tag{1}$$

where the inputs x, x^+, and x^k can be images, point clouds, or patches. The input x^+ is a positive pair of x, and x^k is a negative sample of x. Their instantiations are dependent on specific pretext tasks. The f_q and f_k are encoder networks, which can be identical, partially shared, or different. $\exp(\cdot)$ maps the extracted representation onto scalar-valued scores, where higher scores indicate higher likelihood. τ denotes the temperature, which controls the strength of penalties on the hard negative samples.

B. Overview of DCPoint

Effective representations of 3-D point clouds must encapsulate both local geometric details and global semantic context. Previous SRL methods of 3-D point clouds predominantly focus on scene- or point-level understanding of 3-D point clouds [19], [20]. In contrast, our SRL approach DCPoint introduces a multiperspective contrastive by simultaneously considering the underlying connections among different components and objects. As shown in Fig. 2, DCPoint comprises three fundamental modules: Data augmentation, point cloud network, and joint optimization. Specifically, data augmentation generates semantic-related pairs, referred to View 1 and View 2 in Fig. 2. These pairs contain distinct perspectives on the original point cloud (indicated by different colors) and serve as the foundation for the subsequent global–local contrast task. Point cloud network consists of an online module and a target module capturing multilevel representations of the input semantic-related pairs simultaneously. This includes point-level representations (H^{t_1} and H^{t_2}) for the local contrast, as well as global-level representations (G^{t_1} and G^{t_2}) and global-level contrastive representations (Z^{t_1} and Z^{t_2}) for the global contrast. Joint optimization is focused on extracting hidden structural and semantic information from the hierarchical representations of point clouds based on our global–local dual contrast modules.

1) Data Augmentation: Let P denote an input point cloud. $P \in \mathbb{R}^{N\times 3}$ is a set of vectors, i.e., $P = \{p_1, p_2, \ldots, p_N\}$. Here, p_i consists of the 3-D Cartesian coordinates of a point, and N denotes the number of points in the point cloud P. We apply two different data augmentation operators $\mathcal{T}_1$ and $\mathcal{T}_2$ on P to produce two augmented views P^{t1} and P^{t2}

$$P^{t_1} = \mathcal{T}_1(P) \in \mathbb{R}^{N_1\times 3}, \quad P^{t_2} = \mathcal{T}_2(P) \in \mathbb{R}^{N_1\times 3} \tag{2}$$

where N_1 is the number of points of P^{t_1} and P^{t_2}. The data augmentation strategies include random translation, scaling, cropping, and cutout (see Section IV-A2 for detailed definition).

2) Point Cloud Network: We use the point cloud network to extract multilevel features from two semantically related point clouds P^{t_1} and P^{t_2}. The point cloud network comprises an online module and a target module. These modules contain a feature encoder, a feature mapping, and a projector. Besides, the online module has a predictor.

With the two semantically related point clouds P^{t_1} and P^{t_2}, the feature encoders of online module and target module (i.e., f^o_{En} and f^t_{En}) aim to extract their point-level feature representations H^{t_1} and H^{t_2}. They are illustrated as follows:

$$H^{t_1} = f^o_{\text{En}}(P^{t_1}) \tag{3}$$

$$H^{t_2} = f^t_{\text{En}}(P^{t_2}) \tag{4}$$

$$f^t_{\text{En}} = MA(f^o_{\text{En}}) \tag{5}$$

where $\text{MA}(\cdot)$ denotes an exponential moving average strategy. If we parameterize f^o_{En} by ξ and f^t_{En} by θ, (5) is represented as $\theta \leftarrow \upsilon\theta + (1-\upsilon)\xi$ in each optimization step, where υ denotes a constant and $\upsilon = 0.99$.

After extracting point-level representations of point clouds, we use representation mapping to capture their global-level representations G^{t_1} and G^{t_2}

$$G^{t_1} = [\max(H^{t_1}), \text{avg}(H^{t_1})]$$
$$G^{t_2} = [\max(H^{t_2}), \text{avg}(H^{t_2})] \tag{6}$$

where max denotes max pooling, and avg denotes average pooling. The results of max pooling and average pooling for point-level representations are concatenated to form the global-level representation of point clouds.

We use learnable nonlinear projectors f^o_{Pro} and f^t_{Pro} to map the global-level representations G^{t_1} and G^{t_2} into the contrast space. It can enhance the performance of point cloud encoders, as discussed in [40]. Furthermore, we adopt the predictor of the online module f^t_{Pre} to avoid the collapsed problem. Overall, the online module and target module derive global-level contrastive representations Z^{t_1} and Z^{t_2}, as shown as follows:

$$Z^{t_1} = f^o_{\text{Pre}}\big(f^o_{\text{Pro}}(G^{t_1})\big) \tag{7}$$

$$Z^{t_2} = f^t_{\text{Pro}}(G^{t_2})) \tag{8}$$

$$f^t_{\text{Pro}} = MA(f^o_{\text{Pro}}). \tag{9}$$

In Section IV-A1, we will present the detailed architecture of the point cloud network.

3) Joint Optimization: Given two augmentation views P^{t_1} and P^{t_2}, their point-level representations (H^{t_1} and H^{t_2}) are optimized with our local contrast module. It enforces structurewise discrimination. In addition, their global-level contrastive representations, Z^{t_1} and Z^{t_2}, are optimized with our global contrast module that enforces instancewise consistency. This joint optimization strategy strengthens feature encoders with the desired properties for a wide range of downstream tasks. In the subsequent subsections, we will describe in detail the formulation of our local contrast module (see Section III-C) and our global contrast module (see Section III-D). We will introduce the overall training objective of our proposed DCPoint in Section III-E.

C. Local Contrast

The existing contrastive SRL methods of point clouds mainly focus on instance- or pointwise representations of objects. Contrasting instance-level representations may overlook the internal structural information of point clouds. While contrasting point-level representations might fail to capture the contextual cues necessary for object recognition. Hence, we propose an additional intermediate level of contrast, i.e., the local level. Intuitively, this level focuses on the relationships between the components of objects, which is essential for object understanding. Moreover, the local structure information can boost the performances of point cloud networks that focus on global information of objects [41].

Similar to other levels of contrastive SRL, the fundamental challenge faced by the local contrast revolves around

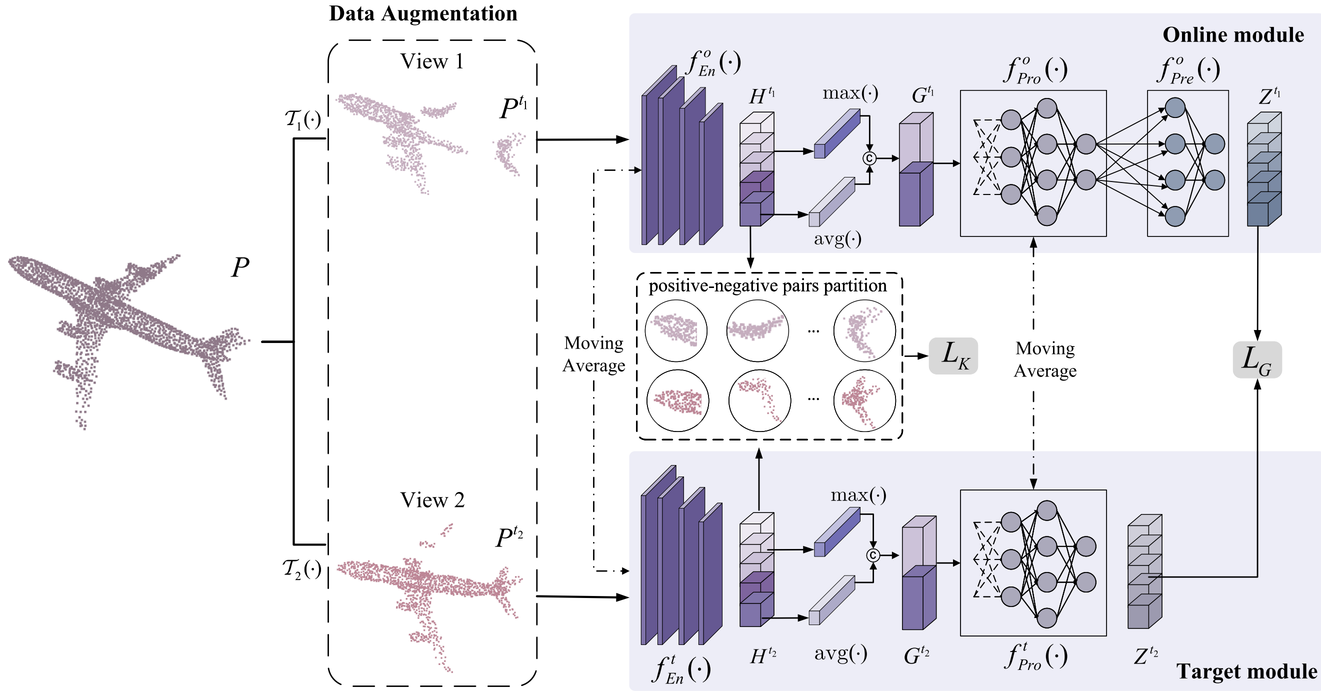


Fig. 2. Illustration of the proposed method DCPoint.

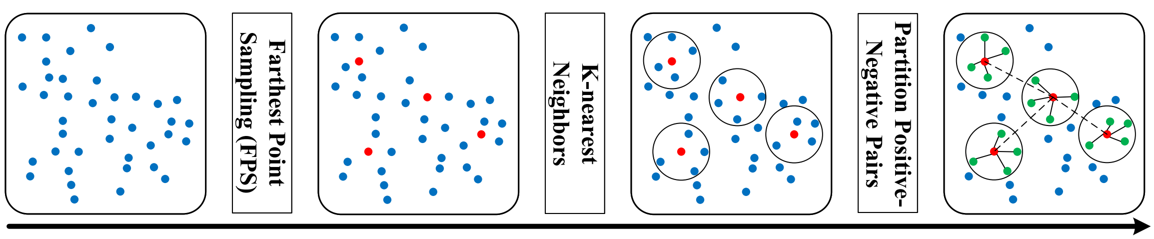


Fig. 3. Illustration of positive–negative pair partitioning based on spatial distribution in our local contrast module.

determining *positive–negative sample pairs*. Previous SRL methods for 2-D vision task [22], [42] divide each image into nonoverlapping grids. They treat the points of each grid as separate instances. It is hereinafter referred to as uniform local contrast. However, it is not straightforward to apply the uniform local contrast to 3-D point clouds due to their sparsity and irregularity. Self-Contrast [12] proposes to pre-train a self-similarity learning model to measure the similarity between different local areas of point clouds. The local regions with high similarity form the positive pairs; otherwise, they form the negative pairs. However, this self-similarity learning model significantly increases the computational complexity.

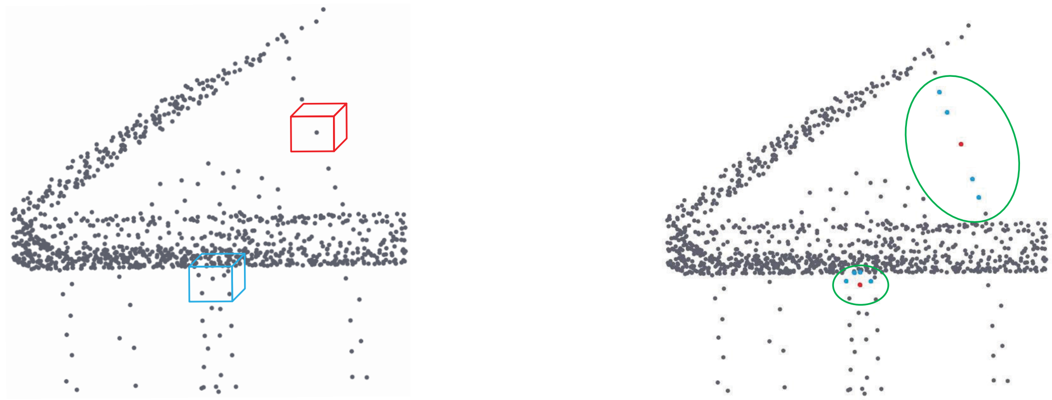
Neighboring points might share the same semantic label and the degree of semantic consistency is related to the distances among points [43], [44]. As such, we propose to define contrastive sample pairs based on the spatial relationships between points. Specifically, as shown in Fig. 3, given a point cloud, we first select some points with farthest point sampling (FPS) algorithm [44] based on their 3-D Cartesian coordinates (i.e., the red points). These selected points can depict the structure of the point cloud to the fullest extent possible. Each selected red point is set as a center and forms a local region with its k-nearest neighbors (i.e., the green points). Each selected red point and its k-nearest neighbors in green form the positive sample pairs (i.e., connected by the solid lines) and a local region. Different centers in red form the negative sample pairs (i.e., connected by the dotted lines). This effective and efficient local region partition strategy is tailored to the unique properties of 3-D point clouds.

Our local contrast module aims to shrink the representation distances between positive sample pairs, promoting feature consistency within local regions. Simultaneously, it enlarges the distances between negative sample pairs, enhancing the discriminative power between distinct components of objects. In summary, we divide each point cloud into the number of C areas. Each area contains the number of $K+1$ points, i.e., a center and its K neighbors. The learning objective of our local contrast is defined as follows:

$$L_K = -\frac{1}{K}\sum_{j=1}^{K}\log\left(\frac{\exp\left({h_i}^T\cdot h_j/\tau\right)}{\sum_o \exp\left({h_i}^T\cdot h_o/\tau\right)}\right) \tag{10}$$

where h_i denotes the representation of a center; h_j denotes the representations of its neighbors within the same local area; and h_o denotes the representations of other centers. τ denotes the temperature, which is set to 0.07 according to [16], [40].

Compared with applying uniform local contrast to point clouds, our pointwise local contrast module effectively avoids



(a) (b)

Fig. 4. Schematic of different local contrast methods. (a) Uniform local contrast [22]. Different colored cubes denote different local areas. (b) Our point-level local contrast. Different red points form negative pairs. The red points and their surrounding blue points form positive pairs. Different green ellipses denote different local areas.

the imperfect and invalid contrast problem. The uniform local contrast module divides point clouds into nonoverlapping cubes with a fixed size, treating each cube as a separate instance. The point cloud for a grand piano is depicted in 4. Observed in Fig. 4(a), the number of points in different cubes varies significantly because of the variations in point cloud sparsity across different regions. The red cube only contains one point of the piano lid support rod, which lacks the corresponding positive pairs. The blue cube contains many points, where positive pairs may include points from the piano legs and keys. It may lead to the imperfect local partition problem. Fig. 4(b) illustrates the partition result of our point-level local contrast module. For the red points of the piano lid support rod, our module determines their surrounding points as the positive pairs, as shown in the larger green ellipse. In the junction of the piano legs and keys, our module determines the most similar points as the positive pairs, as shown in the smaller green ellipse. As such, the sparsity does not impact the stability of our pointwise local contrast module.

D. Global Contrast

Global contrastive SRL methods learn the semantic relationships among unlabeled objects by constraining their global-level contrastive representations. It has been reported the favorable performance in both 2-D and 3-D domains [16], [17]. In global contrastive SRL methods, the positive pairs contain different augmented views of objects. The negative pairs contain different object instances from a mini-batch. However, this selection strategy might generate imperfect negative pairs. For example, a negative pair may comprise different instances of the same category. It can result in erroneous feature distribution after enlarging embedding distances between samples in the negative pair. To ensure the reliable contrast, our global contrast module omits negative pairs. However, learning only from positive pairs may result in collapsed problems, i.e., models derive the same output vector for all inputs. To mitigate the risk of convergence issues, our DCPoint implements global contrast by facilitating interactions of two asymmetric modules: the online and target modules. Z^{t_1} and Z^{t_2} denote the outputs of the online module and target module, respectively. The primary learning objective of our global contrast mechanism is to minimize the discrepancy between Z^{t_1} and Z^{t_2}, which is quantified with the Euclidean metric. The learning objective of our global contrast is defined as follows:

$$L_G = \left\| Z^{t_1} - Z^{t_2} \right\|_2^2. \tag{11}$$

E. Global–Local Joint Objective

Our proposed DCPoint incorporates the local contrast loss function in addition to the global contrast loss function for joint optimization. It is to simultaneously support the contrast properties of global semantic and local structural information of 3-D point clouds. The global–local joint objective is defined as follows:

$$L = L_G + \alpha L_K \tag{12}$$

where α is a balancing coefficient, ensuring a balanced order of magnitudes among different constraint functions. Our dual contrast method does not incur additional overhead for feature computation compared with that with only using global contrast. Algorithm 1 provides the pseudocode of the proposed DCPoint.

IV. Implementation and Experiments

A. Implementation Details

1) Architecture: As shown in Fig. 2, our DCPoint consists of the feature encoders (f^o_{En} and f^t_{En}), the projectors (f^o_{Pro} and f^t_{Pro}), and the predictor (f^o_{Pre}). The feature encoders capture pointwise features of point clouds to be used by our local contrast module. The feature encoder of DGCNN [43] has been widely applied in various 3-D vision tasks. We select it as the default feature encoders of DCPoint. In addition, we adopt the feature encoder of CurveNet [45] as the feature encoders of DCPoint to evaluate the feasibility of DCPoint using different feature encoders.

The projectors of DCPoint contain two fully connected (FC) layers. The first FC layer projects the global features of objects into 4096 dimensions. It is followed by batch normalization

Algorithm 1 Pseudocode of DCPoint

initialize
f^t_{En}.params $= f^o_{En}$.params
f^t_{Pro}.params $= f^o_{Pro}$.params
# load a point cloud P
for P in loader:
    # generate two different augmented views
    $P^{t_1} = \mathcal{T}_1(P), P^{t_2} = \mathcal{T}_2(P)$ # (2)
    # capture the point-level representation
    $H^{t_1} = f^o_{En}(P^{t_1})$ # (3)
    $H^{t_2} = f^t_{En}(P^{t_2})$ # (4)
    # capture the global-level representation
    $G^{t_1} = f_g(H^{t_1}), G^{t_2} = f_g(H^{t_2})$ # (6)
    # capture the global-level contrastive representation
    $Z^{t_1} = f^o_{Pre}(f^o_{Pro}(G^{t_1}))$ # (7)
    $Z^{t_2} = f^t_{Pro}(G^{t_2})$ # (8)
    # partition positive-negative pairs for local contrast
    $\tilde{H}^{t_1} = pn(H^{t_1}), \tilde{H}^{t_2} = pn(H^{t_2})$ # Fig. 3
    # Local contrast loss
    $loss_l = L_K(\tilde{H}^{t_1}, \tilde{H}^{t_2})$ # (10)
    # Global contrast loss
    $loss_g = L_G(Z^{t_1}, Z^{t_2})$# (11)
    # Global-Local joint loss
    $loss = loss_g + \alpha loss_l$ # (12)
    # parameters update: online module
    $loss$.backward()
    update(f^o_{En}, f^o_{Pro}, f^o_{Pre})
    # momentum update: target module
    $f^t_{En} = MA(f^o_{En})$, $f^t_{Pro} = MA(f^o_{Pro})$ # (5), (9)

and rectified linear units (ReLUs). The second FC layer projects the output of the first FC layer into 256 dimensions.

The predictor is exclusively used for the online module of DCPoint. It is to predict the output of the target module, preventing collapse in an unsupervised scenario [23]. The predictor is similar to the projector, but the dimensions of their input data are different.

In the global contrast module of DCPoint, we generate two augmented views of a point cloud through the same augmentation methods used in STRL [20]. In the local contrast module of DCPoint, we divide each point cloud into 512 local areas, where each local area contains a center point and four nearest points. These hyperparameters will be discussed in the ablation studies represented in Section IV-D3.

2) Point Cloud Augmentation Operations: We first sample point clouds with different strategies for different downstream tasks. The sampling details are represented in Section IV-A4. To obtain the semantic-corrected pair of each point cloud, we augment each sampled point cloud twice with a set of geometric transformation operations, such as random translation (shifted within [0, 0.05]), scaling ([0.8, 1.2]), cropping ([0.75, 1.33]), and cutout ([0.1, 0.4]).

3) Optimization: We design the two-stage optimization strategy for pretraining models with our DCPoint. In the first stage, we train models with our global contrast module using (11). In the second stage, we continue to train these models with our local contrast module using (12). The coefficient α is set to 0.01 empirically.

Our proposed architecture is implemented on the PyTorch platform. The optimizer is the Adam combined with layerwise adaptive rate scaling (LARS) and the cosine decay learning rate schedule. In the first stage, we train the models for 100 epochs on two NVIDIA GeForce RTX 3090 with a batch size of 32. The initial learning rate is set to $1e^{-3}$. In the second stage, we set the initial learning rate to $1e^{-6}$ with a batch size of 5 on a single NVIDIA GeForce RTX 2080Ti for five epochs.

4) Datasets for Pretraining: To be consistent with previous works [16], [20], we pretrain models with our proposed DCPoint on the datasets as follows:

1) *ShapeNet*[1]*:* We pretrain DCPoint on the ShapeNet dataset [53] for the downstream classification and part segmentation tasks. ShapeNet consists of 57 448 point clouds of 55 categories. In applications, we randomly sample 2048 points from each point cloud.
2) *ScanNet*[2]*:* We pretrain DCPoint on the ScanNet dataset [54] for the downstream semantic segmentation tasks. As an RGB-D video dataset, ScanNet consists of 1513 scenes from 707 real-world indoor environments. We subsample the raw videos at a periodic interval (by default, once every 100 frames). Therefore, we get a subset of ScanNet, which consists of 24 902 frames. To obtain the point cloud from a given RGB-D frame, we transfer the locations of pixels (u, v) in an RGB-D frame to 3-D points (X, Y, Z) with the camera intrinsics M using the following equation:

$$Z\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = M\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}. \tag{13}$$

In the experiments, we randomly sample 4096 points from each projected point cloud.

B. Downstream Tasks

We evaluate the transferability of DCPoint on three widely used downstream tasks in 3-D SRL: 1) 3-D object classification with linear evaluation, fine-tuning, and FSL; 2) 3-D part segmentation with semi-supervised learning; and 3) 3-D semantic segmentation with semi-supervised learning.

1) 3-D Object Classification:

a) ModelNet40[3]*:* As a widely used synthetic point cloud dataset, ModelNet40 [24] contains 12 311 samples of 3-D computer-aided design (CAD) over 40 common object categories. Among them, 9843 samples are for training, and the remaining 2468 samples are for testing.

[1]https://shapenet.org/
[2]http://www.scan-net.org/
[3]https://shapenet.cs.stanford.edu/media/modelnet40_normal_resampled.zip

TABLE I
THREE-DIMENSIONAL OBJECT CLASSIFICATION WITH LINEAR EVALUATION ON MODELNET40

Method	Publication	Year	SRL Category	Accuracy (%)
Latent-GAN [46]	PMRL	2018	Generative	85.7
SO-Net [47]	CVPR	2018	Generative	87.3
FoldingNet [48]	CVPR	2018	Generative	88.4
MRTNet [49]	ECCV	2018	Generative	86.4
3D-PointCapsNet [50]	CVPR	2019	Generative	88.9
Multi-Task [35]	ICCV	2019	Generative	89.1
VIP-GAN [51]	AAAI	2019	Generative	90.2
Jigsaw [36]	NIPS	2019	Generative	90.6
DepthContrast [52]	ICCV	2021	Context	85.4
OcCo [37]	ICCV	2021	Generative	89.2
Self-Contrast [12]	ACMMM	2021	Context	89.6
STRL [20]	ICCV	2021	Context	90.9
CrossPoint [16]*	CVPR	2022	Context	91.2
Point-MAE [14]	ECCV	2022	Generative	91.2
ACT [39]*	ICLR	2023	Generative	91.4
DCPoint(Ours)		2023	Context	**91.5**

”*” the model simultaneously uses the multi-modal information of objects, such as 3D point clouds, 2D images, and 1D natural languages.

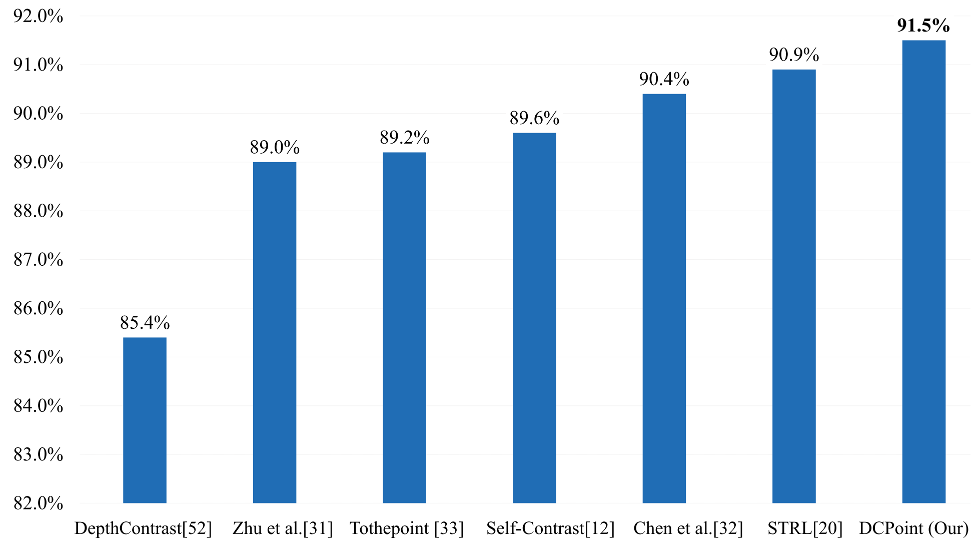


Fig. 5. Three-dimensional object linear classification with unimodal contrastive SRL method on ModelNet40.

b) ScanObjectNN[4]*:* As a popular real-world point cloud dataset, ScanObjectNN [25] contains 2902 scanned samples over 15 categories. About 80% of these samples are used for training, and the rest are used for testing. To ensure a fair comparison, we use the same dataset as CrossPoint [16].

c) Object classification with linear evaluation: To demonstrate the generalizability of our proposed DCPoint on the 3-D object classification, we evaluate the classification accuracy of our model with linear classification heads. The corresponding evaluation metric is shown as follows:

$$\text{Accuracy} = \frac{C_a}{C_N} \times 100\% \tag{14}$$

[4]https://hkust-vgd.github.io/scanobjectnn/

where C_N denotes the total number of testing samples, and C_a denotes the number of samples correctly classified by a model.

In the implementation, we integrate a linear support vector machine (SVM) classifier with the pretrained feature encoder to form a classification model. We fine-tune the SVM parameters throughout the training process while keeping the pretrained feature encoder parameters frozen. During the testing phase, we assess the performance of classification models, wherein the feature encoders are pretrained using our DCPoint method or previous SRL methods. Table I, Figs. 5, and 6 present the results of these models on the ModelNet40 and ScanObjectNN datasets.

As shown in Table I, DCPoint achieves a linear classification accuracy of 91.5% on ModelNet40. In comparison to multimodal SRL methods [16], [39], which use the knowledge

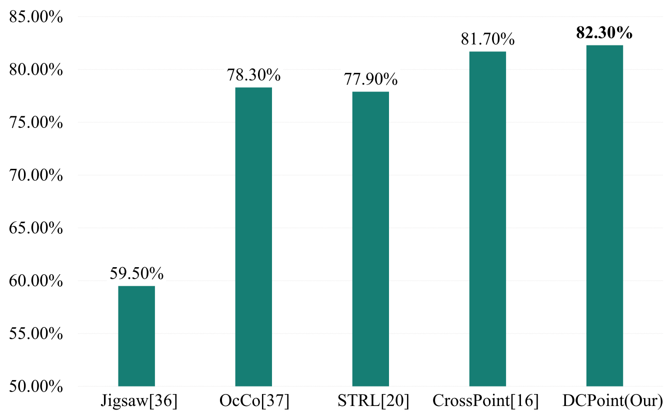


Fig. 6. Three-dimensional object classification with linear evaluation on ScanObjectNN.

of 2-D images and 1-D natural language to guide SRL of 3-D point clouds, our DCPoint demonstrates competitive performance by constraining the representation distribution of point clouds from multiperspective.

In Fig. 5, we further present the comparative results of unimodal contrastive SRL methods. DCPoint outperforms global contrast SRL methods by significant margins. Specifically, it surpasses Zhu et al. [31] by 2.5%, Tothepoint [33] by 2.3%, and STRL [20] by 0.6%. In addition, DCPoint demonstrates superior performance compared with the voxel-point global contrastive method DepthContrast [52] by 6.1% and the local contrastive SRL method Self-Contrast [12] by 1.9%. Furthermore, DCPoint exceeds the performance of Chen et al. [32] by 1.1%, a method that combines resolution recovery and global contrast tasks to learn intrinsic feature representations. These experimental findings underscore the enhanced semantic awareness exhibited by point cloud encoders, which capture multilevel information of objects during the pretraining phase.

Fig. 6 shows the classification results of our proposed DCPoint and other SRL works on ScanObjectNN. Compared with ModelNet40, ScanObjectNN contains more complex background noises. Therefore, all the previous works and our DCPoint achieve lower accuracies on ScanObjectNN than those on ModelNet40. In such cases, the accuracy of DCPoint surpasses previous SRL methods, e.g., DCPoint outperforms the multimodal contrastive SRL method CrossPoint by 0.6%. These experimental results verify DCPoint's generalization and effectiveness for out-of-distribution data.

d) Object classification with FSL: FSL trains models with limited data. It is commonly used to test the generalization of SRL methods [8]. In the training stage, models are optimized with $N \times K$ samples over N categories (hereinafter called N-way K-shot). In the FSL experiments of our DCPoint, we randomly select the training samples and use the same testing samples in different trials. The final results of the models are the mean and standard deviation of their classification accuracies over ten replications. The classification models in our FSL experiments consist of an SVM classifier and feature encoders, which are pretrained by different SRL methods. Table II shows the experimental results on the ModelNet40 and ScanObjectNN datasets with FSL.

It is seen in Table II that the proposed DCPoint outperforms other SRL models on the ModelNet40 and ScanObjetNN datasets. It is worth noting that DCPoint is less affected by the scale of training data than other methods. The mean accuracy of CrossPoint in the ten-way ten-shot experiments is 8.9% lower than of the five-way ten-shot experiments on ModelNet40. While the mean accuracy of DCPoint only decreases 1.8% in the same experiments. This is because our global–local dual contrast method captures more essential features of 3-D objects by simultaneously learning the distinctions between the inter- and intraobjects. However, the previous contrastive SRL methods ignore the relationships between interobjects, and the previous generative SRL methods ignore the relationships between intraobjects. In addition, DCPoint consistently outperforms its randomly initialized counterpart, DGCNN, by significant margins in various FSL experiments. For example, the mean accuracy gain is up to 55% on ModelNet40 and 15.5% on ScanObjectNN in the five-way 20-shot experiments.

e) Object classification with fine-tuning: We also evaluate our SRL method DCPoint by supervised fine-tuning. In the training step, the pretrained model provides the initial weights for the feature encoder of the point cloud classifier. The parameters of the point cloud classifier are optimized with all the training samples of classification datasets. Table III shows the fine-tuned results of our DCPoint and previous SRL methods on ModelNet40 and ScanObjectNN. All the SRL models share the same architecture, i.e., DGCNN. Compared with the randomly initialized DGCNN, DCPoint achieves a performance increase of 0.7% on ModelNet40 and 3.5% on ScanObjectNN. These improvements are more significant than the previous SRL methods.

2) 3-D Object Part Segmentation:

a) ShapeNetPart[5]: As a popular part segmentation dataset for 3-D point clouds, ShapeNetPart [26] contains 16 881 samples (14 007 for training and 2874 for testing) over 16 object categories and 50 part categories.

b) Semi-supervised learning: In the experiments of part segmentation with semi-supervised learning, we first pretrain the feature encoders of DGCNN with our DCPoint and STRL [20] on the ShapeNet dataset. Then, we fine-tune DGCNN with a small percentage of training data (e.g., 1%–10%) of ShapeNetPart for 200 epochs with a batch size of 32. The optimizer is a standard SGD with a momentum of 0.9. The initial learning rate is set to $1e^{-3}$. To evaluate the segmentation performance of DGCNN, we use the mean intersection over union (mIoU) as the evaluation metric, as denoted in (15). All the experiments are based on the PyTorch platform with one NVIDIA GeForce RTX 2080Ti

$$\text{mIoU} = \frac{1}{|\mathcal{C}|} \sum_{c=1}^{\mathcal{C}} \frac{\left|\{y = c\} \cap \{\tilde{y} = c\}\right|}{\left|\{y = c\} \cup \{\tilde{y} = c\}\right|} \tag{15}$$

where $\mathcal{C}$ denotes a finite set of classes, c denotes one of the categories, y denotes the pointwise ground-truth labels, and $\tilde{y}$ denotes the predicted pointwise results.

[5]https://shapenet.org/

TABLE II
THREE-DIMENSIONAL OBJECT CLASSIFICATION WITH FSL ON MODELNET40 AND SCANOBJECTNN. THE RESULTS ARE THE MEAN AND STANDARD ERROR OVER TEN REPLICATIONS

Method	Publication	Year	5-way		10-way	
			10-shot	20-shot	10-shot	20-shot
ModelNet40						
Latent-GAN [46]	PMRL	2018	41.6 ±5.3	46.2±6.2	32.9±2.9	25.5±3.2
FoldingNet [48]	CVPR	2018	33.4 ±4.1	35.8±5.8	18.6±1.8	15.4±2.2
DGCNN [43]	TOG	2019	31.6 ±2.8	40.8±4.6	19.9±2.1	16.9±1.5
3D-PointCapsNet [50]	CVPR	2019	42.3 ±5.5	53.0±5.9	38.0±4.5	27.2±4.7
Jigsaw [36]	NIPS	2019	34.3 ±1.3	42.2±3.5	26.0±2.4	29.9±2.6
OcCo [37]	ICCV	2021	90.6 ±2.8	92.5±1.9	82.9±1.3	86.5±2.2
CrossPoint [16]*	CVPR	2022	92.5 ±3.0	94.9±2.1	83.6±5.3	87.9±4.2
Point-MAE [14]	ECCV	2022	91.1 ± 5.6	91.7±4.0	83.5±6.1	89.7±4.1
ACT [39]*	ICLR	2023	91.8 ±4.7	93.1±4.2	84.5±6.4	90.7±4.3
DCPoint(Ours)		2023	**92.6±4.2**	**95.8±3.0**	**90.8±1.8**	**92.7±1.0**
ScanObjectNN						
DGCNN [43]	TOG	2019	62.0 ±5.6	67.8±5.1	37.8±4.3	41.8±2.4
Jigsaw [36]	NIPS	2019	65.2 ±3.8	72.2±2.7	45.6±3.1	48.2±2.8
OcCo [37]	ICCV	2021	72.4 ±1.4	77.2±1.4	57.0±1.3	61.6±1.2
CrossPoint [16]*	CVPR	2022	74.8 ±1.5	79.0±1.2	62.9±1.7	73.9±2.2
DCPoint(Ours)		2023	**75.0±5.8**	**83.3±3.6**	**65.6±4.3**	**75.0±4.4**

”*” the model simultaneously uses the multi-modal information of objects, such as 3D point clouds, 2D image, and 1D natural language.

TABLE III
THREE-DIMENSIONAL OBJECT CLASSIFICATION WITH FINE-TUNING ON MODELNET40 AND SCANOBJECTNN

Method	SRL Category	Accuracy (%)	
		ModelNet40	ScanObjectNN
DGCNN [43]	-	92.5	82.4
Jigsaw [36]	Generative	92.3 (-0.2)	82.7 (+0.3)
OcCo [37]	Generative	93.0 (+0.5)	83.9 (+1.5)
STRL [20]	Context-based	93.1 (+0.6)	85.4 (+3.0)
DCPoint(Ours)	Context-based	**93.2 (+0.7)**	**85.9 (+3.5)**

”-” the baseline model, which is random initialization without any pretraining stages;
”( )” the improvement of SRL method over the baseline model.

TABLE IV
THREE-DIMENSIONAL PART SEGMENTATION WITH SEMI-SUPERVISED LEARNING ON SHAPENETPART. PERCENTAGE DENOTES THE PERCENTAGE OF TRAINING DATA IN THE TRAINING SET

Percentage	SRL Method	mIoU (%)
1%	Baseline model	75.1
	STRL [20]	74.1 (-1.0)
	DCPoint (ours)	**75.4 (+0.3)**
10%	Baseline model	81.3
	STRL [20]	81.6 (+0.3)
	DCPoint (ours)	**81.8 (+0.5)**

”( )” the improvement of SRL method over the baseline model.

As shown in Table IV, when fine-tuning with 1% of the training data from the ShapeNetPart dataset, DCPoint outperforms its baseline model counterpart by 0.3% of mIoU, which is random initialized without any pretraining stages. But STRL performs 1.0% worse than the randomly initialized counterpart. As the fine-tuning data increase to 10%, our DCPoint outperforms STRL by 0.2% and the randomly initialized counterpart by 0.5%. These experimental results indicate the significance of our local contrast module in the part segmentation task. It highlights the necessity for point cloud encoders to extract local point-level features.

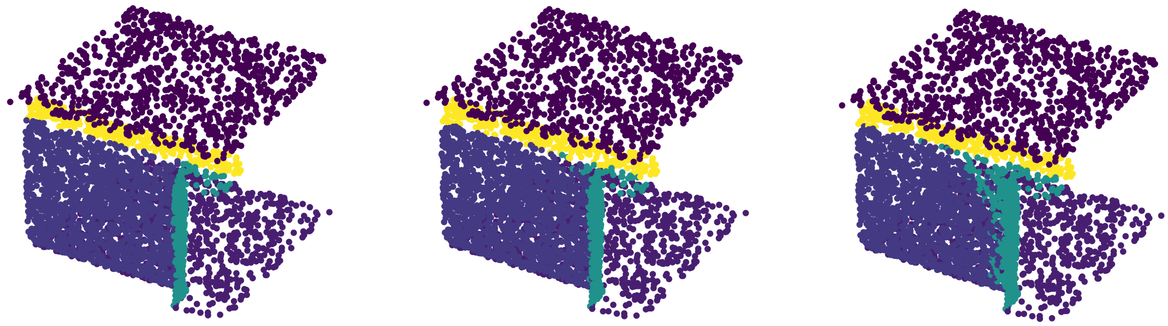
3) 3-D Object Semantic Segmentation:

a) S3DIS[6]*:* As a large-scale point cloud dataset of indoor spaces, S3DIS [27] contains 3-D scanned data from six large-scale indoor areas, denoted as Area 1–Area 6, with 695 878 620 points over 13 categories. Following the previous work [20], we sample point clouds of each room by selecting the key points within an area 1×1 m and randomly resample 4096 points from each sampled point cloud.

b) Semi-supervised learning: In this experiment of semantic segmentation with semi-supervised learning, we first pretrain the feature encoders of DGCNN with our DCPoint and STRL [20] on the ScanNet dataset. Then, we fine-tune DGCNN with Area 1–Area 5 of S3DIS and test it on Area 6. In the fine-tuning stage, we use a standard SGD optimizer with momentum 0.9. The batch size is 32, and the total fine-tuning is 100 epochs. The initial learning rate is set to $1e^{-3}$. All the experiments are based on the PyTorch platform with one NVIDIA GeForce RTX 2080Ti.

As shown in Table V, DCPoint consistently outperforms its randomly initialized baseline counterpart. In particular, when fine-tuning with Area 3, which only has 1640 samples, DCPoint outperforms the randomly initialized baseline model by 1.8% and achieves better results than STRL. When fine-tuning with Area 5, which has 6852 samples, DCPoint outperforms the randomly initialized baseline counterpart by

[6]http://buildingparser.stanford.edu/dataset.html



(a) (b) (c)

Fig. 7. Segmentation results on Area 6 of S3DIS. (a) Ground truth. (b) DCPoint (Our). (c) STRL [20].

TABLE V
THREE-DIMENSIONAL SEMANTIC SEGMENTATION WITH SEMI-SUPERVISED LEARNING ON S3DIS

Area for training	SRL Method	mIoU on Area 6 (%)
	Baseline model	57.3
Area 1 (3687 samples)	STRL [20]	56.9 (-0.4)
	DCPoint (ours)	**58.0 (+0.7)**
	Baseline model	37.8
Area 2 (4440 samples)	STRL [20]	38.4 (+0.6)
	DCPoint (ours)	**38.9 (+1.1)**
	Baseline model	49.1
Area 3 (1650 samples)	STRL	50.7 (+1.6)
	DCPoint (ours)	**50.9 (+1.8)**
	Baseline model	36.5
Area 4 (3662 samples)	STRL [20]	37.1 (+0.6)
	DCPoint (ours)	**37.4 (+0.9)**
	Baseline model	47.2
Area 5 (6852 samples)	**STRL** [20]	**49.3 (+2.1)**
	DCPoint (ours)	**49.3 (+2.1)**

"( )" the improvement of SRL method over the baseline model.

2.1% and performs similar to STRL. Fig. 7 shows the segmentation results on Area 6 of DCPoint and STRL fine-tuning with Area 1. It is clear that the most significant discrepancies between DCPoint and STRL locate in the junctions between three local areas. DCPoint obtains more accurate segmentation results than STRL. The cause of the performance superiority is that our local contrast module guides the feature encoder to learn the local details. Furthermore, DCPoint exhibits greater accuracy than STRL when labeled data are minimal. This indicates that DCPoint captures more general fine-grained architecture attributes of 3-D objects, which is essential in cross-domain semantic segmentation.

C. Further Analysis of DCPoint

1) Generality of Local Contrast: The previous experiments show that DCPoint performs much better on different 3-D downstream tasks. In this section, we perform 3-D object linear classification and FSL experiments to investigate the generality of our local contrast module. Specifically, we evaluate the performances of combining our local contrast module with other SRL methods, including STRL [20] and self-orientation [28], i.e., STRL + local contrast and self-orientation + local contrast. Among them, STRL pretrains the models by comparing the global features of objects; self-orientation pretrains the models by predicting the orientation of objects. For a fair comparison, we leverage the publicly available source codes of STRL and self-orientation. In the pretraining process, we first train DGCNN using these previous SRL methods. Next, we retrain it using our local contrast module. We use ShapeNet as the pretraining dataset and verify the model performances on ModelNet and ScanObjectNN.

As shown in Table VI, after incorporating with our local contrast module, the linear classification accuracy of self-orientation is improved by 0.7% on ModelNet40 and 1.0% on ScanObjectNN. The self-orientation + local contrast always outperforms self-orientation in various FSL experiments. For instance, in the five-way ten-shot experiments, the mean accuracy gain is increased by 1.4% on ModelNet40 and 1.8% on ScanObjectNN. The accuracy of STRL + local contrast is higher than that of STRL by 0.6% in the linear classification experiments and 2.2% in the five-way ten shot experiments on ModelNet40. Although STRL + local contrast only slightly improves the accuracy compared with STRL in the FSL experiments on ScanObjectNN, it outperforms STRL by 4.4% in the linear classification. The reason is that our local contrast module enhances the local feature extraction capability of STRL. It can achieve larger improvements when fine-tuning with more training data on a complex real-world dataset.

D. Ablation Studies of DCPoint

1) Architecture of Feature Encoder: In this section, we perform 3-D object classification with FSL experiments to investigate the generality of DCPoint on different feature encoders. We select the feature encoders of two models, including DGCNN [43] and CurveNet [45]. These models are graph-based feature extraction networks. The graph of DGCNN is created based on nearby points in a small region, whereas a continuous sequence of nonlocal points forms the graph of CurveNet. We pretrain these feature encoders using our DCPoint and STRL [20] on ShapeNet. We compare their performance in the FSL experiments on ModelNet40.

Table VII shows that our DCPoint outperforms STRL in the FSL experiments regardless of the feature encoder. These results confirm that DCPoint can be applied to various feature encoders to capture more general object features. It is notable that the feature encoder of DGCNN using SRL methods could

TABLE VI
THREE-DIMENSIONAL LINEAR CLASSIFICATION AND FSL RESULTS ON MODELNET40 AND SCANOBJECTNN. THE RESULTS ARE THE MEAN AND STANDARD ERROR OVER TEN REPLICATIONS IN FSL EXPERIMENTS

Pre-train Method	Linear Classification (%)	Accuracy of Few-Shot Learning (%)			
		5-way		10-way	
		10-shot	20-shot	10-shot	20-shot
ModelNet40					
Self-Orientation [28]	87.6	88.2±5.5	89.4±5.7	86.4±5.8	88.2±5.7
Self-Orientation + Local Contrast	**88.3 (+0.7)**	**89.6±3.4**	**90.4±4.3**	**87.5±6.0**	**88.5+5.4**
STRL [20]	90.9	90.4±4.8	95.5±2.5	84.9±3.4	91.8±2.5
STRL + Local Contrast	**91.5 (+0.6)**	**92.6±4.2**	**95.8±3.0**	**90.8±1.8**	**92.7±1.0**
ScanObjectNN					
Self-Orientation [28]	63.9	74.5±7.6	75.5±7.6	73.5±8.7	72.8±8.5
Self-Orientation + Local Contrast	**64.9 (+1.0)**	**76.3±5.3**	**78.8±6.0**	**74.0±8.6**	**74.1±7.9**
STRL [20]	77.9	74.6±7.0	82.8±4.8	65.2±4.4	73.5+5.0
STRL + Local Contrast	**82.3 (+4.4)**	**75.0±5.8**	**83.3±3.6**	**65.6±4.3**	**75.0±4.4**

”( )” the relative gain achieved by our local contrast module.

TABLE VII
ABLATION OF THE FEATURE ENCODER OF SRL METHODS

Feature encoder	SRL method	5-way		10-way	
		10-shot	20-shot	10-shot	20-shot
DGCNN [43]	STRL [20]	90.4±4.8	95.5±2.5	84.9±3.4	91.8±2.5
	DCPoint (Ours)	**92.6±4.2**	**95.8±3.0**	**90.8±1.8**	**92.7±1.0**
CurveNet [45]	STRL [20]	91.5±5.2	95.0±2.5	88.2±2.2	92.3±1.9
	DCPoint (Ours)	**92.9±4.9**	**95.2±6.5**	**89.3±3.5**	**92.7±2.9**

TABLE VIII
ABLATION OF DIFFERENT CONTRAST METHODS. OUR DEFAULT SETTINGS ARE SHOWN IN GRAY

Model	Contrast Category	One-stage optimization	Two-stage optimization	Accuracy (%)
A	Global-local Contrast	✓		89.5
B	**Global-local Contrast**		✓	**91.5**
C	Global Contrast	✓		90.9
D	Global Contrast		✓	90.8
E	Local Contrast	✓		85.0
F	Local Contrast		✓	85.5

TABLE IX
ABLATION OF HYPERPARAMETERS. OUR DEFAULT SETTINGS ARE SHOWN IN GRAY (a) NUMBER OF LOCAL AREAS C OF A POINT CLOUD. (THE NUMBER OF NEIGHBOR POINTS K OF A CENTER POINT IS SET TO 4.) (b) NUMBER OF NEIGHBORS K OF A CENTER POINT. (THE NUMBER OF LOCAL AREAS C OF A POINT CLOUD IS SET TO 512)

(a)

C	Accuracy (%)
128	91.0
256	91.2
512	**91.5**
1024	91.2

(b)

K	Accuracy (%)
2	91.0
4	**91.5**
16	91.1
32	90.0

perform better than the feature encoder of CurveNet in some FSL experiments. The related FSL literature [55] has reported that complex networks might degrade FSL performances.

2) Global–Local Dual Contrast Versus Global Contrast Versus Local Contrast: In this section, we design detailed studies to illustrate the effectiveness of our proposed global–local dual contrast method. Specifically, we compare it to the cases of only using global or local contrast. To pretrain the feature encoder of DGCNN, we use different contrast methods on ShapeNet with different optimization strategies, such as one-stage and two-stage optimization strategies. As its name implies, the one-stage optimization strategy only contains one training process. The two-stage optimization strategy contains two training processes. The second training process starts from the parameters learned by the first training process. After pretraining, we compare their 3-D object linear classification accuracies on ModelNet40. Table VIII shows the experimental results.

a) Global–local dual contrast with different optimization strategies: Different optimization strategies can bring different performances even under the same model. As shown in Table VIII, Model A denotes DGCNN pretrained with the global–local dual contrast under the one-stage optimization, which obtains a classification accuracy of 89.5% and is lower than Model B by 2%. The two-stage optimization of Model B means that the model is first trained with the global contrast and then trained with the global–local dual contrast. After such an incremental optimization strategy, the model can realize more complex learning objectives.

b) Global–local dual contrast versus global contrast: As shown in Table VIII, Model B outperforms the global contrast (Model C) by 0.6%. Model B is equivalent to adding the local contrast to Model C in the second optimization stage. To further verify the pertinence between the performance improvement and our local contrast module, we retrain Model C with global contrast by the same optimization strategy as the second training stage of Model B, i.e., Model D. However, the performance of Model D is lower than Model C. The reason is that directly retraining Model C leads to overfit. While retraining with our local contrast helps improve the model's generalization.

c) Global–local dual contrast versus local contrast: As shown in Table VIII, Model E is pretrained only with the local contrast and gets the lowest classification accuracy of 85.0%. Model F has a two-stage optimization strategy. In the first stage, it is pretrained with global contrast. In the second stage, it is pretrained with local contrast. Model F obtains a classification accuracy of 85.5%. The reason is that the local contrast ignores the invariance between different instances, which is vital to classification tasks.

3) Point Sampling: Our proposed local contrast module of point clouds aims to keep the consistency of the center point and its neighbors within a local area. It aims to enlarge the differences between the center points of different local areas. Therefore, the number of neighbors K of a center point and the number of local areas C are essential to our local contrast module. We ablate such hyperparameters in the 3-D object linear classification experiments on ModelNet40.

As shown in Table IX(a), if the value of K is set as 4, changes in the value of C will not significantly impact the results. However, if the value of C is kept to 512, the model's performance starts to saturate with $K = 4$, as shown in Table IX(b). The reason is that the more the neighbors of a center point, the weaker the correlations between the center point and its neighbors. It leads to incorrect guidance for representation learning of point clouds.

V. Conclusion

This article introduces DCPoint, a global–local dual contrastive SRL method for 3-D point clouds. Its global contrast module aims to capture the instance-level characteristics of objects by minimizing the distance between the two augmented inputs in the global representation space. The local contrast module of DCPoint aims to capture the detailed characteristics of objects by enhancing interpartition consistency and intrapartition discrimination on the pointwise representation space. Tailored to the unique properties of 3-D point clouds, the partitioning of positive and negative pairs for the local contrast is dependent on their spatial distribution. Therefore, DCPoint enables the simultaneous learning of internal structural and semantic characteristics of objects. In the downstream tasks, such as 3-D object classification and segmentation in synthetic and real-world datasets, DCPoint outperforms its randomly initialized baseline counterparts and previous SRL methods. This article highlights the importance of multiperspective contrastive learning for 3-D point clouds, which holds great potential for advancing related studies. Moreover, the proposed local contrast module can further improve the performances of other SRL methods.

In future work, we plan to investigate a one-stage optimization strategy for DCPoint to improve its training efficiency. In addition, we aim to explore the extension of our multiperspective contrastive strategy to multimodality SRL.

References

[1] H. Wang, J. Xu, Y. Huang, G. Zhang, Y. Rong, and W. Yu, "Multilayer positioning strategy for tubesheet welding robot based on point cloud model," *IEEE Sensors J.*, vol. 23, no. 12, pp. 13728–13737, Jun. 2023.

[2] B. Tan et al., "3D object detection for multi-frame 4D automotive millimeter-wave radar point cloud," *IEEE Sensors J.*, 2022.

[3] Q. Gao, Y. Chen, Z. Ju, and Y. Liang, "Dynamic hand gesture recognition based on 3D hand pose estimation for human–robot interaction," *IEEE Sensors J.*, vol. 22, no. 18, pp. 17421–17430, Sep. 2022.

[4] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. Comput. Vis. Pattern Recognit.*, 2016.

[5] L. Lai, J. Chen, C. Zhang, Z. Zhang, G. Lin, and Q. Wu, "Tackling background ambiguities in multi-class few-shot point cloud semantic segmentation," *Knowl.-Based Syst.*, vol. 253, Oct. 2022, Art. no. 109508.

[6] M. Zhao et al., "PCUNet: A context-aware deep network for coarse-to-fine point cloud completion," *IEEE Sensors J.*, vol. 22, no. 15, pp. 15098–15110, Aug. 2022.

[7] X. Wang, Y. Jin, Y. Cen, T. Wang, B. Tang, and Y. Li, "LighTN: Light-weight transformer network for performance-overhead tradeoff in point cloud downsampling," *IEEE Trans. Multimedia*, early access, Sep. 22, 2024, doi: 10.1109/TMM.2023.3318073.

[8] A. Xiao, J. Huang, D. Guan, X. Zhang, S. Lu, and L. Shao, "Unsupervised point cloud representation learning with deep neural networks: A survey," 2022, *arXiv:2202.13589*.

[9] X. Long, Z. Zhang, and Y. Li, "Multi-network contrastive learning of visual representations," *Knowl.-Based Syst.*, vol. 258, Dec. 2022, Art. no. 109991.

[10] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 16000–16009.

[11] C. Tao, J. Qi, M. Guo, Q. Zhu, and H. Li, "Self-supervised remote sensing feature learning: Learning paradigms," *IEEE Trans. Geosci. Remote Sens.*, 2023.

[12] B. Du, X. Gao, W. Hu, and X. Li, "Self-contrastive learning with hard negative sampling for self-supervised point cloud learning," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 3133–3142.

[13] C. Sun, Z. Zheng, X. Wang, M. Xu, and Y. Yang, "Self-supervised point cloud representation learning via separating mixed shapes," *IEEE Trans. Multimedia*, pp. 1–11, 2022.

[14] Y. Pang, W. Wang, F. E. H. Tay, W. Liu, Y. Tian, and L. Yuan, "Masked autoencoders for point cloud self-supervised learning," 2022, *arXiv:2203.06604*.

[15] X. Yu, L. Tang, Y. Rao, T. Huang, J. Zhou, and J. Lu, "Point-BERT: Pre-training 3D point cloud transformers with masked point modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 19313–19322.

[16] M. Afham, I. Dissanayake, D. Dissanayake, A. Dharmasiri, K. Thilakarathna, and R. Rodrigo, "Crosspoint: Self-supervised cross-modal contrastive learning for 3D point cloud understanding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 9902–9912.

[17] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9729–9738.

[18] X. Liu et al., "Self-supervised learning: Generative or contrastive," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 1, pp. 857–876, Jan. 2023.

[19] S. Xie, J. Gu, D. Guo, C. R. Qi, L. Guibas, and O. Litany, "Point-Contrast: Unsupervised pre-training for 3D point cloud understanding," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 574–591.

[20] S. Huang, Y. Xie, S.-C. Zhu, and Y. Zhu, "Spatio-temporal self-supervised representation learning for 3D point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 6535–6545.

[21] O. Shrout, O. Nitzan, Y. Ben-Shabat, and A. Tal, "PatchContrast: Self-supervised pre-training for 3D object detection," 2023, *arXiv:2308.06985*.

[22] Y. Bai, X. Chen, A. Kirillov, A. Yuille, and A. C. Berg, "Point-level region contrast for object detection pre-training," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 16040–16049.

[23] J.-B. Grill et al., "Bootstrap your own latent-a new approach to self-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 21271–21284.

[24] Z. Wu et al., "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1912–1920.

[25] M. A. Uy, Q. Pham, B. Hua, T. Nguyen, and S. Yeung, "Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1588–1597.

[26] L. Yi et al., "A scalable active framework for region annotation in 3D shape collections," *ACM Trans. Graph.*, vol. 35, no. 6, pp. 1–12, Nov. 2016.

[27] I. Armeni et al., "3D semantic parsing of large-scale indoor spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1534–1543.

[28] O. Poursaeed, T. Jiang, H. Qiao, N. Xu, and V. G. Kim, "Self-supervised learning of point clouds via orientation estimation," in *Proc. Int. Conf. 3D Vis. (3DV)*, Nov. 2020, pp. 1018–1028.

[29] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 4037–4058, Nov. 2021.

[30] R. Dangovski et al., "Equivariant self-supervised learning: Encouraging equivariance in representations," in *Proc. Int. Conf. Learn. Represent.*, 2021.

[31] F. Zhu, J. Zhao, and Z. Cai, "A contrastive learning method for the visual representation of 3D point clouds," *Algorithms*, vol. 15, no. 3, p. 89, Mar. 2022.

[32] H. Chen, S. Luo, X. Gao, and W. Hu, "Unsupervised learning of geometric sampling invariant representations for 3D point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 893–903.

[33] X. Li, J. Chen, J. Ouyang, H. Deng, S. Velipasalar, and D. Wu, "ToThePoint: Efficient contrastive learning of 3D point clouds via recycling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 21781–21790.

[34] Z. Li et al., "SimIPU: Simple 2D image and 3D point cloud unsupervised pre-training for spatial-aware visual representations," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, 2022, pp. 1500–1508.

[35] K. Hassani and M. Haley, "Unsupervised multi-task feature learning on point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8159–8170.

[36] J. Sauder and B. Sievers, "Self-supervised deep learning on point clouds by reconstructing space," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019.

[37] H. Wang, Q. Liu, X. Yue, J. Lasenby, and M. J. Kusner, "Unsupervised point cloud pre-training via occlusion completion," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9762–9772.

[38] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.

[39] R. Dong et al., "Autoencoders as cross-modal teachers: Can pre-trained 2D image transformers help 3D representation learning?" in *Proc. 11th Int. Conf. Learn. Represent.*, 2023. [Online]. Available: https://openreview.net/forum?id=8Oun8ZUVe8N

[40] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. 37th Int. Conf. Mach. Learn.*, Jul. 2020, pp. 1597–1607.

[41] Z. Huang, Z. Zhao, B. Li, and J. Han, "LCPFormer: Towards effective 3D point cloud analysis via local context propagation in transformers," *IEEE Trans. Circuits Syst. Video Technol.*, 2023.

[42] X. Wang, R. Zhang, C. Shen, T. Kong, and L. Li, "Dense contrastive learning for self-supervised visual pre-training," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 3024–3033.

[43] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, Oct. 2019.

[44] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.

[45] T. Xiang, C. Zhang, Y. Song, J. Yu, and W. Cai, "Walk in the cloud: Learning curves for point clouds shape analysis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 915–924.

[46] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3D point clouds," in *Proc. 35th Int. Conf. Mach. Learn.*, vol. 80, Jul. 2018, pp. 40–49.

[47] J. Li, B. M. Chen, and G. H. Lee, "SO-Net: Self-organizing network for point cloud analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9397–9406.

[48] Y. Yang, C. Feng, Y. Shen, and D. Tian, "FoldingNet: Point cloud auto-encoder via deep grid deformation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 206–215.

[49] M. Gadelha, R. Wang, and S. Maji, "Multiresolution tree networks for 3D point cloud processing," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 103–118.

[50] Y. Zhao, T. Birdal, H. Deng, and F. Tombari, "3D point capsule networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1009–1018.

[51] Z. Han, M. Shang, Y.-S. Liu, and M. Zwicker, "View inter-prediction GAN: Unsupervised representation learning for 3D shapes by learning global shape memories to support local view predictions," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 8376–8384.

[52] Z. Zhang, R. Girdhar, A. Joulin, and I. Misra, "Self-supervised pretraining of 3D features on any point-cloud," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 10252–10263.

[53] A. X. Chang et al., "ShapeNet: An information-rich 3D model repository," 2015, *arXiv:1512.03012*.

[54] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "ScanNet: Richly-annotated 3D reconstructions of indoor scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5828–5839.

[55] Y. Tian, Y. Wang, D. Krishnan, J. B. Tenenbaum, and P. Isola, "Rethinking few-shot image classification: A good embedding is all you need?" in *Proc. 16th Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 266–282.

Lu Shi received the master's degree in computer applications technology from Xi'an Technological University, Shaanxi, China, in 2021. She is currently pursuing the Ph.D. degree with the Institute of Information Science and Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing Jiaotong University, Beijing, China.

Her research interests include computer vision and 3-D processing.

Guoqing Zhang received the bachelor's degree in software engineering from Linyi University, Shandong, China, in 2021. He is currently pursuing the Ph.D. degree with the Institute of Information Science, Beijing Jiaotong University, Beijing, China.

His research interests include semantic segmentation, scene graph generation, and multimodality.

Qi Cao received the B.Eng. degree from Huazhong University of Science Technology (HUST), Wuhan, China, in 2000, and the Ph.D. degree from Nanyang Technological University, Singapore, in 2007.

He is currently an Assistant Professor with the School of Computing Science, University of Glasgow, Singapore. His research interests include computational intelligence, virtual reality, image processing, and data analytics.

Linna Zhang received the bachelor's degree in mechanical design and manufacturing from Guizhou University of Technology, Guiyang, China, in 2000, and the master's degree in mechanical engineering from Guizhou University, in 2010.

From September 2019 to August 2020, she was a Visiting Scholar at the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China. Her research interests include computer vision.

Yigang Cen received the Ph.D. degree in control science engineering from Huazhong University of Science Technology, Wuhan, China, in 2006.

In 2006, he joined the Signal Processing Centre, School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, as a Research Fellow. From 2014 to 2015, he was a Visiting Scholar at the Department of Computer Science, University of Missouri, Columbia, MO, USA. He is currently a Professor and a Supervisor of doctoral students with the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China. His research interests include computer vision, multimedia understanding, and intelligent transportation.

Yi Cen received the B.Eng. degree from the School of Electronic Information and Communication, Huazhong University of Science and Technology, Wuhan, China, in 2008, and the Ph.D. degree in engineering from the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China, in 2014.

Since 2014, he has been teaching in Minzu University of China, Beijing. His research interests include computer vision and multimedia understanding.