4DContrast: Contrastive Learning with Dynamic Correspondences for 3D Scene Understanding

Yujin Chen, Matthias Nießner, and Angela Dai

Technical University of Munich
{yujin.chen,niessner,angela.dai}@tum.de

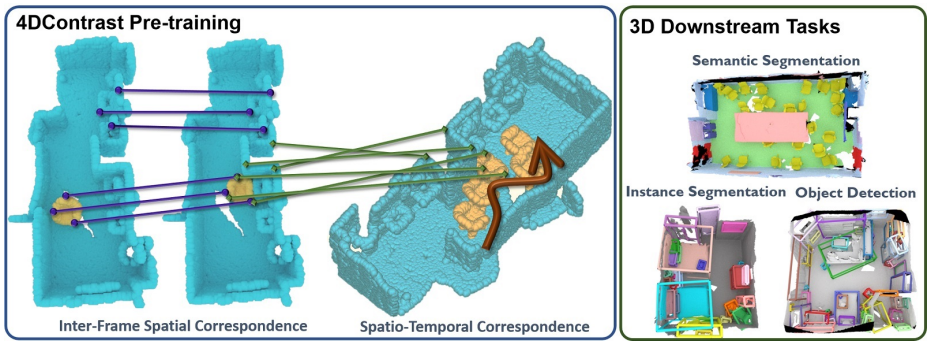


Fig. 1. We propose 4DContrast to imbue learned 3D representations with 4D priors. We introduce a data augmentation scheme to composite synthetic 3D objects with real-world 3D scans to create 4D sequence data with inherent correspondence information. We then leverage a combination of 3D-3D, 3D-4D, and 4D-4D constraints within a contrastive learning framework to learn 4D invariance in the 3D representations. The learned features can be transferred to improve performance in various downstream 3D scene understanding tasks.

Abstract. We present a new approach to instill 4D dynamic object priors into learned 3D representations by unsupervised pre-training. We observe that dynamic movement of an object through an environment provides important cues about its objectness, and thus propose to imbue learned 3D representations with such dynamic understanding, that can then be effectively transferred to improved performance in downstream 3D semantic scene understanding tasks. We propose a new data augmentation scheme leveraging synthetic 3D shapes moving in static 3D environments, and employ contrastive learning under 3D-4D constraints that encode 4D invariances into the learned 3D representations. Experiments demonstrate that our unsupervised representation learning results in improvement in downstream 3D semantic segmentation, object detection, and instance segmentation tasks, and moreover, notably improves performance in data-scarce scenarios. Our results show that our 4D pre-training method improves downstream tasks such as object detection mAP@0.5 by 5.5%/6.5% over training from scratch on ScanNet/SUN RGB-D while involving no additional run-time overhead at test time.

Keywords: 3D scene understanding, point cloud recognition, 3D semantic segmentation, 3D instance segmentation, 3D object detection

1 Introduction

3D semantic scene understanding has seen remarkable progress in recent years, in large part driven by advances in deep learning as well as the introduction of large-scale, annotated datasets [1,7,10]. In particular, notable progress has been made to address core 3D scene understanding tasks such as 3D semantic segmentation, object detection, and instance segmentation, which are fundamental to many real-world computer vision applications such as robotics, mixed reality, or autonomous driving. Such approaches have developed various methods to learn on different 3D scene representations, such as sparse or dense volumetric grids [6,7,11], point clouds [29,30], meshes [19], or multi-view approaches [8,38]. Recently, driven by the success of unsupervised representation learning for transfer learning in 2D, 3D scene understanding methods have been augmented with unsupervised 3D pre-training to further improve performance to downstream 3D scene understanding tasks [17,20,43,46].

While such 3D representation learning has focused on feature representations learned from static 3D scenes, we observe that important notions of objectness are given by 4D dynamic observations – for instance, object segmentations can often be naturally intuited by observing objects moving around an environment without any annotations required, which can be more difficult in a static 3D observation. We thus propose to leverage this powerful 4D signal in unsupervised pre-training to imbue 4D object priors into learned 3D representations, that can then be effectively transferred to various downstream 3D scene understanding tasks for improved recognition performance.

In this work, we introduce 4DContrast to learn about objectness from both static 3D and dynamic 4D information in learned 3D representations. We leverage a combination of static 3D scanned scenes and a database of synthetic 3D shapes, and augment the scenes with moving synthetic shapes to generate 4D sequence data with inherent motion correspondence. We then employ a contrastive learning scheme under both 3D and 4D constraints, correlating local 3D point features with each other as well as with 4D sequence features, thus imbuing learned objectness from dynamic information into the 3D representation learning.

To demonstrate our approach, we pre-train on ScanNet [7] along with ModelNet [41] shapes for unsupervised 3D representation learning. Experiments on 3D semantic segmentation, object detection, and instance segmentation show that 4DContrast learns effective features that can be transferred to achieve improved performance in various downstream 3D scene understanding tasks. 4DContrast can also generalize from pre-training on ScanNet and ModelNet to improved performance on SUN RGB-D [36]. Additionally, we show that our learned representations remain robust in limited training data scenarios, consistently improving performance under a various amounts of training data available.

Our main contributions are summarized as follows:

- We propose the first method to leverage 4D sequence information and constraints for 3D representation learning, showing transferability of the learned

features to the downstream 3D scene understanding tasks of 3D semantic segmentation, object detection, and instance segmentation.
– Our new unsupervised pre-training based on constructing 4D sequences from synthetic 3D shapes in real-world, static 3D scenes improves performance across a variety of downstream tasks and different datasets.

2 Related Work

3D Semantic Scene Understanding. Driven by rapid developments in deep learning and the introduction of several large-scale, annotated 3D datasets [1,7,10], notable progress has been made in 3D semantic scene understanding, in particular the tasks of 3D semantic segmentation [6,7,8,11,18,19,24,30,32], 3D object detection [25,26,27,42,47], and 3D instance segmentation [9,13,16,21,45]. Many methods have been proposed, largely focusing on learning on various 3D representations, such as sparse or dense volumetric grids [6,7,11], point clouds [21,27,29,30], meshes [19,35], or multi-view hybrid representations [8,22]. In particular, approaches leveraging backbones built with sparse convolutions [6,11] have shown strong effectiveness across a variety of 3D scene understanding tasks and datasets. We propose a new unsupervised pre-training approach to learn 4D priors in learned 3D representations, leveraging sparse convolutional backbones for both 3D and 4D feature extraction.
3D Representation Learning. Inspired by the success of representation learning in 2D, particularly that leveraging instance discrimination with contrastive learning [2,3,15], recent works have explored unsupervised learning with 3D pretext tasks that can be leveraged for fine-tuning on downstream 3D scene understanding tasks [5,14,17,20,23,31,33,34,39,40,43,46]. For instance, [14,34] learn feature representations from point-based instance discrimination for object classification and segmentation, and [17,43,46] extend to more complex 3D scenes by generating correspondences from various different views of scene point clouds.

In particular, given the more scarce data availability of real-world 3D environments, Hou et al. [17] additionally demonstrate the efficacy of contrastive 3D pretraining for various 3D semantic scene understanding tasks under a variety of limited training data scenarios. In contrast to these methods that employ 3D-only pretext tasks for representation learning, we propose to learn from 4D sequence data to embed 4D priors into learned 3D representations for more effective transfer to downstream 3D tasks.

Recently, Huang et al. [20] propose to learn from the inherent sequence data of RGB-D video to incorporate the notion of a temporal sequence. Constraints are established across pairs of frames in the sequence; however, the sequence data itself represents static scenes without any movement within the scene, limiting the temporal signal that can be learned. In contrast, we consider 4D sequence data containing object movement through the scene, which can provide additional semantic signal about objectness through an object's motion. Additionally, Rao et al. [31] propose to learn from 3D scenes that are synthetically generated by randomly placing synthetic CAD models on a rectangular layout.

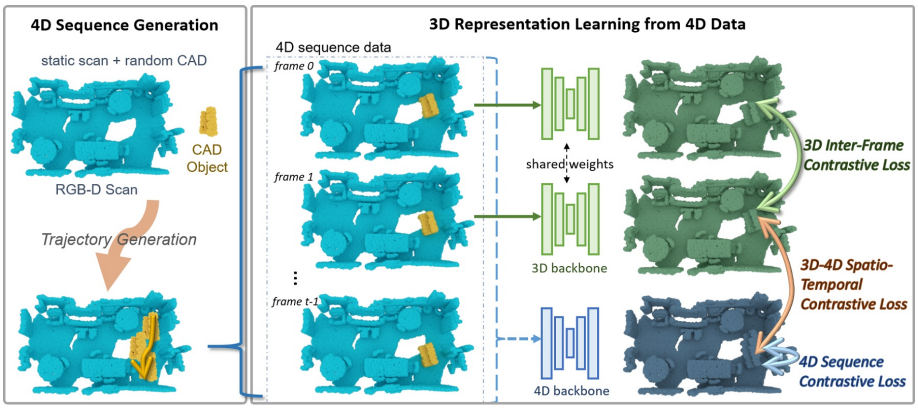


Fig. 2. Method overview. 4DContrast learns effective 3D feature representations imbued with 4D signal from moving object sequences. During pre-training, we augment static 3D scene data with a moving object from a synthetic shape dataset. We can then establish dynamic correspondences between the spatio-temporal features learned from the 4D sequence with 3D features of individual static frames. We employ contrastive learning under not only 3D geometric correspondence between individual frames, but also with their corresponding 4D counterpart, as well as 4D-4D constraints to anchor the 4D feature learning. This enables 4D-invariant representation learning, which we can apply to various downstream 3D scene understanding tasks.

They employ object-level contrastive learning on object-level features, resulting in improved 3D object detection performance. We also leverage synthetic CAD models for data augmentation, but we compose them with real-world 3D scan data to generate 4D sequences of objects in motion, and exploit learned 4D features to enhance learned 3D representations, with performance improvement on various downstream 3D scene understanding tasks.

3 4D Invariant Representation Learning

4DContrast presents a new approach to 3D representation learning: our key idea is to employ 4D constraints during pre-training, in order to imbue learned features with 4D invariance from learned objectness from seeing an object in motion. We consider a dataset of 3D scans $\mathcal{S} = \{S_i\}$ as well as a dataset of synthetic 3D objects $\mathcal{O} = \{O_j\}$, and construct dynamic sequences with inherent correspondence information by moving a synthetic object O_j in a static 3D scan S_i. This enables us to establish 4D correspondences along with 3D-4D correspondence as constraints under a contrastive learning framework for unsupervised pre-training. An overview of our approach is shown in Figure 2.

3.1 Revisiting SimSiam

We first revisit SimSiam [4], which introduced a simple yet powerful approach for contrastive 2D representation learning. Inspired by the effectiveness of SimSiam,

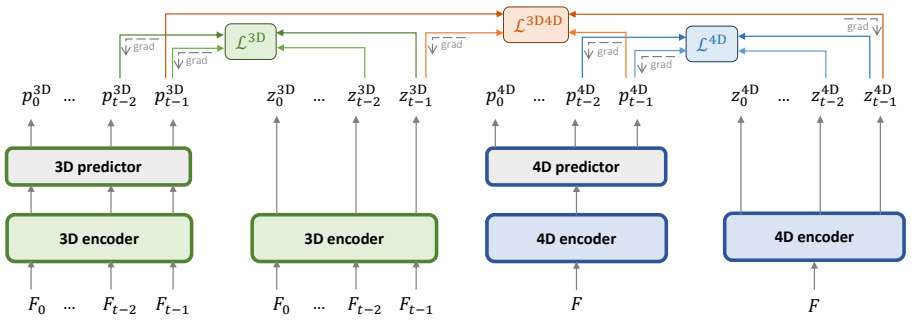


Fig. 3. 4DContrast pre-training. We visualize 3D-3D, 3D-4D, and 4D-4D losses across frame and spatio-temporal correspondence. Note that losses are established across all pairs of frames for $\mathcal{L}^{3D}$ $\mathcal{L}^{4D}$ and across all frames for $\mathcal{L}^{3D4D}$; for visualization we only show those associated with frames F_{t-2} and F_{t-1}, and only for F_{t-1} for $\mathcal{L}^{3D4D}$. Each loss only propagates back according to the gradient arrows due to stop-gradient operations for stable training.

we build an unsupervised contrastive learning scheme for embedding 4D priors into 3D representations.

SimSiam considers two augmented variants of an image I, I_1, and I_2, which are input to weight-shared encoder network Φ_{2D} (a 2D convolutional backbone followed by a projection MLP). Then a prediction MLP head P_{2D} transforms the output of one view as $p_1^{2D} = P_{2D}(\Phi_{2D}(I_1))$ to match to the another output $z_2^{2D} = \Phi_{2D}(I_2)$, with minimizing the negative cosine similarity [12]:

$$\mathcal{D}(p_1^{2D}, z_2^{2D}) = -\frac{p_1^{2D}}{||p_1^{2D}||_2} \cdot \frac{z_2^{2D}}{||z_2^{2D}||_2}. \tag{1}$$

SimSiam also uses a stop-gradient (SG) operation that treats z_2^{2D} as a constant during back-propagation, to prevent collapse during the training, thus modifying Eq. 1 as: $\mathcal{D}(p_1^{2D}, SG(z_2^{2D}))$. A symmetrized loss is defined for the two augmented inputs:

$$\mathcal{L}^{2D} = \frac{1}{2}\mathcal{D}(p_1^{2D}, SG(z_2^{2D})) + \frac{1}{2}\mathcal{D}(p_2^{2D}, SG(z_1^{2D})). \tag{2}$$

SimSiam has shown to be very effective at learning invariances under various image augmentations, without requiring negative samples or very large batches. We thus build from this contrastive framework for our 3D-4D constraints, as it allows for our high-dimensional pre-training design.

3.2 4D-Invariant Contrastive Learning

To imbue effective 4D priors into learned 3D features, we consider a static 3D scan S and a synthetic 3D object O as a train sample, and compose them together to form dynamic object movement in the scene $\{F_0, ..., F_{t-1}\}$ for t time steps (as described in Section 3.3). We then establish spatial correspondences between frames (3D-3D), spatio-temporal correspondences (3D-4D), and dynamic correspondences (4D-4D) as constraints. 3D features are extracted with a 3D

encoder Φ_{3D} and 4D features with a 4D encoder Φ_{4D}, with respective prediction MLPs P_{3D} and P_{4D}.

Inter-Frame Spatial Correspondence. For each pair of frames (F_i, F_j) in a train sequence F, we consider their spatial correspondence across sequence frames in order to implicitly pose invariance over the dynamic sequence. That is, points that correspond to the same location in the original 3D scene S or original object O should also correspond in feature space. For the set of corresponding point locations $\mathcal{A}_{i,j}$ from frames (F_i, F_j), we consider each pair of point locations $(\mathbf{a}_i, \mathbf{b}_j) \in \mathcal{A}$, we obtain their 3D backbone features at the respective locations: $p^{3D}_{i,a} = P_{3D}(\Phi_{3D}(F_i))(\mathbf{a}_i)$ and $z^{3D}_{j,b} = \Phi_{3D}(F_j)(\mathbf{b}_j)$. We then compute a symmetrized negative cosine similarity loss between features of corresponding point locations:

$$\mathcal{L}^{3D}_{\mathcal{A}_{i,j}} = \sum_{(a,b)\in\mathcal{A}_{i,j}} \left(\frac{1}{2}\mathcal{D}(p^{3D}_{i,a}, SG(z^{3D}_{j,b})) + \frac{1}{2}\mathcal{D}(p^{3D}_{j,b}, SG(z^{3D}_{i,a})) \right). \tag{3}$$

In Figure 3, we use green arrows to indicate constraints between frame F_{t-2} and frame F_{t-1}.

We compute Eq. 3 over each pair of frames in the whole sequence F:

$$\mathcal{L}^{3D} = \sum_{\substack{i=0 \\ i<j}}^{t-1} \sum_{j=0}^{t-1} \mathcal{L}^{3D}_{\mathcal{A}_{i,j}}. \tag{4}$$

By establishing constraints across 3D frames in a 4D sequence, we encode pose invariance of moving objects across varying background into the learned 3D features.

Spatio-Temporal Correspondence. In addition to implicitly encoding pose invariance of moving objects, we establish explicit 3D-4D correspondences to learn 4D priors, encouraging 4D-invariance in the learned features. For a train sequence $F = \{F_0, ..., F_{t-1}\}$, we use the 4D encoder Φ_{4D} and the 4D predictor P_{4D} to extract 4D features from the whole sequence. Then $z^{4D}_{i,a}$ indicates the 4D features output by the 4D encoder Φ_{4D} at point location $\mathbf{a}_i$ in frame i, and $p^{4D}_{i,a}$ denotes the 4D features output by the 4D predictor P_{4D}. Then for a frame F_i, we consider each 3D point $\mathbf{a}_i \in \mathcal{A}_i$ in this set of frame points F_i, and establish a constraint between its corresponding 3D feature extracted by 3D network (Φ_{3D} and P_{3D}) and its corresponding 4D feature extracted by 4D network (Φ_{4D} and P_{4D}):

$$\mathcal{L}^{3D4D}_{\mathcal{A}_i} = \sum_{a\in\mathcal{A}_i} \left(\frac{1}{2}\mathcal{D}(SG(p^{3D}_{i,a}), z^{4D}_{i,a}) + \frac{1}{2}\mathcal{D}(SG(p^{4D}_{i,a}), z^{3D}_{i,a}) \right). \tag{5}$$

As shown in Figure 3, we use orange arrows to indicate constraints of frame F_{t-1}. For the entire input sequence F, we calculate Eq. 5 for every frame, and the 3D-4D contrastive loss $\mathcal{L}^{3D4D}$ is defined as:

$$\mathcal{L}^{3D4D} = \sum_{i=0}^{t-1} \mathcal{L}^{3D4D}_{\mathcal{A}_i}. \tag{6}$$

Additionally, in order to learn spatio-temporally consistent 4D representations, we employ 4D-4D correspondence constraints inherent to the 4D features within the same point cloud sequence. This is formulated analogously to Eq. 3, replacing the 3D features with the 4D features from different time steps that correspond spatially:

$$\mathcal{L}^{\text{4D}}_{\mathcal{A}_{i,j}} = \sum_{(a)\in\mathcal{A}_{i,j}} \left(\frac{1}{2}\mathcal{D}(p^{\text{4D}}_{i,a}, SG(z^{\text{4D}}_{j,a})) + \frac{1}{2}\mathcal{D}(p^{\text{4D}}_{j,a}, SG(z^{\text{4D}}_{i,a})) \right). \tag{7}$$

In Figure 3, we use blue arrows to indicate 4D constraints between frame F_{t-2} and frame F_{t-1}. We evaluate Eq. 7 over every pair of frames in the entire input sequence F, with the 4D contrastive loss $\mathcal{L}^{\text{4D}}$ defined as:

$$\mathcal{L}^{\text{4D}} = \sum_{\substack{i=0 \\ i<j}}^{t-1} \sum_{j=1}^{t-1} \mathcal{L}^{\text{4D}}_{\mathcal{A}_{i,j}}. \tag{8}$$

Joint Learning. Our overall training loss $\mathcal{L}$ consists of three parts including 3D contrastive loss $\mathcal{L}^{\text{3D}}$, 3D-4D contrastive loss $\mathcal{L}^{\text{3D4D}}$, and 4D contrastive loss $\mathcal{L}^{\text{4D}}$:

$$\mathcal{L} = w_{\text{3D}}\mathcal{L}^{\text{3D}} + w_{\text{3D4D}}\mathcal{L}^{\text{3D4D}} + w_{\text{4D}}\mathcal{L}^{\text{4D}}, \tag{9}$$

where constant weights w_{3D}, w_{3D4D} and w_{4D} are used to balance the losses.

3.3 Generating 4D Correspondence via Scene-Object Augmentation

To learn from 4D sequence data to embed 4D priors into learned 3D representations, we leverage existing large-scale real-world 3D scan datasets in combination with synthetic 3D shape datasets. This enables generation of 4D correspondences without requiring any labels – by augmenting static 3D scenes with generated trajectories of a moving synthetic object within the scene, which provides inherent 4D correspondence knowledge across the object motion. Thus for pre-training, we consider pairs of reconstructed scans and an arbitrarily sampled synthetic 3D shape (S, O), and generate a 4D sequence $F = \{F_0, ..., F_{t-1}\}$ by moving the object through the scene.

Trajectory Generation. We first generate a trajectory for O in S. We voxelize S at 10cm voxel resolution, and accumulate occupied surface voxels in the height dimension to acquire a 2D map of the scene geometry. Valid object locations are then identified as those in the 2D map with a voxel accumulation ≤ 1, with the max height of the accumulated voxels near to the ground floor (within 20cm of the average floor height). For the object O, we consider all possible 2D locations, and if O does not exceed the valid region (based on its bounding sphere), then the location is taken as a candidate object position. A random position sampled from these candidate positions is taken as the starting point of the trajectory, we can randomly sample a step distance in $[30, 90]$cm and step direction such that the angular change in trajectory is $< 150°$, and then select the nearest

Table 1. Summary of fine-tuning of 4DContrast for various downstream 3D scene understanding tasks and datasets. Our pre-training approach learns effective, transferable features, resulting in notable improvement over the baseline learning paradigm of training from scratch.

Datasets	Stats (#train / #val)	Task	Gain (from scratch)
ScanNetV2 [7]	1.2K / 312 scenes	Sem. Seg.	**+2.3%** mIoU
		Ins. Seg.	**+4.2%** mAP@0.5
		Obj. Det.	**+5.5%** mAP@0.5
SUN RGB-D [36]	5.2K / 5K frames	Obj. Det.	**+6.5%** mAP@0.5

valid candidate position as the second trajectory point. We repeat this process for t time steps in the sequence to obtain 4D scene-object augmentations for pre-training.

4D Sequence Generation. A sequence of point clouds are then generated based on the computed object trajectory for the scan, up to sequence length t, by compositing the object into the scene under its translation and rotation steps per frame. This provides inherent correspondence information between 3D scene locations and 4D object movement through the scene.

Scene Augmentation. We augment the 4D sequences by randomly sampling different points across the geometry in each individual frame. We also randomly remove cubic chunks of points in the background 3D scene for additional data variation, with the number of chunks removed randomly sampled from $[5, 15]$ and the size of the chunks randomly sampled in $[0.15, 0.45]$ as a proportion of the scene extent. We discard any sequences that do not have enough correspondences in its frames; that is, $\geq 30\%$ of the points in the original scan and $\geq 30\%$ of the points of the synthetic object should be consistently represented in each frame, and each frame must maintain at least 50% of its points through the augmentation process. Additionally, we further augment the static 3D frame interpretations of the sequence (but not the sequence) by applying random rotation, translation, and scaling to each individually considered 3D frame.

3.4 Network Architecture for Pre-training

During pre-training, we leverage correspondences induced by our 4D data generation, between encoded 3D frames as well as across the encoded 4D sequence. To this end, we employ 3D and 4D feature extractors as meta-architectures for 4D-invariant learning.

To extract per-point features from a 3D scene, we use a 3D encoder Φ_{3D} and a 3D predictor P_{3D}. Φ_{3D} is a U-Net architecture based on sparse 3D convolutions with residual block followed by a $1 \times 1 \times 1$ sparse convolutional projection layer, and P_{3D} is two $1 \times 1 \times 1$ sparse convolutional layers.

To extract spatio-temporal features from a 4D sequence, we use a 4D encoder Φ_{4D} and a 4D predictor P_{4D}. These are structured analogously to the 3D feature extraction, using sparse 4D convolutions instead. For more detailed architecture specifications, we refer to the supplemental material.

4 Experimental Setup

We demonstrate the effectiveness of our 4D-informed pre-training of learned 3D representations for a variety of downstream 3D scene understanding tasks.

Pre-training Setup. We use reconstructed 3D scans from ScanNet [7] and synthetic 3D shapes from ModelNet [41] to compose our 4D sequence data for pre-training. We use the official ScanNet train split with 1201 train scans, augmented with shapes from ModelNet from eight furniture categories: chair, desk, dresser, nightstand, sofa, table, bathtub, and toilet. For each 3D scan, we generate 20 trajectories of an object moving through the scan, following Section 3.3 with $t = 4$. For sequence generation we use 2cm resolution for the scene and 1000 randomly sampled points from the synthetic object to compose together.

The 3D and 4D sparse U-Nets are implemented with MinkowskiEngine [6] using 2cm voxel size for 3D and 5cm voxel size for 4D. For pre-training we consider only geometry information from the scene-object sequence augmentations. We use an SGD optimizer with initial learning rate 0.25 and a batch-size of 12. The learning rate is decreased by a factor of 0.99 every 1000 steps. We train for 50K steps until convergence.

Fine-tuning on Downstream Tasks. We use the same pre-trained backbone network in the three 3D scene understanding tasks of semantic segmentation, instance segmentation, and object detection. For semantic segmentation, we directly use the U-Net architecture for dense label prediction, and for object detection and instance segmentation, we use VoteNet [27] and PointGroup [21] respectively, both with our pre-trained 3D U-Net backbone. All experiments, including comparisons with state of the art, are trained with geometric information only, unless otherwise noted. Fine-tuning experiments on semantic segmentation are trained with a batch size of 48 for 10K steps, using an initial learning rate of 0.8 with polynomial decay with power 0.9. For instance segmentation, we use the same training setup as PointGroup, and use an initial learning rate of 0.1. For object detection, the network is trained for 500 epochs, and the learning rate is 0.001 and decayed by a factor of 0.5 at epochs 250, 350, and 450. We use a batch size of 6 on ScanNet and 16 on SUN RGB-D.

Table 2. 3D object detection on ScanNet. Our 4DContrast pre-training leads to improved performance in comparison with state of the art object detection and 3D pre-training schemes.

Method	Input	mAP@0.5
DSS [37]	Geo + RGB	6.8
F-PointNet [28]	Geo + RGB	10.8
GSPN [44]	Geo + RGB	17.7
3D-SIS [16]	Geo + RGB	22.5
VoteNet [27]	Geo + Height	33.5
Scratch + VoteNet	Geo only	34.5
RandomRooms [31] + VoteNet	Geo only	36.2 (+1.7)
PointContrast [43] + VoteNet	Geo only	38.0 (+3.5)
CSC [17] + VoteNet	Geo only	39.3 (+4.8)
Ours + VoteNet	Geo only	**40.0** **(+5.5)**

5 Results

We demonstrate that our learned features under 3D-4D constraints can effectively transfer well to a variety of downstream 3D scene understanding tasks. We consider both in-domain transfer to 3D scene understanding tasks on ScanNet [7] (Section 5.1), as well as out-of-domain transfer to SUN RGB-D [36] (Section 5.2); a summary is shown in Table 1. We also show data-efficient scene understanding (Section 5.3) and additional analysis (Section 5.4). Note that for all downstream experiments, we do not use the 4D backbone and thus use the same 3D U-Net architecture as PointContrast [43] and CSC [17].

All experiments, including our method and all baseline comparisons, are trained on geometric data only without any color information.

5.1 ScanNet

We first demonstrate our 4DContrast pre-training in fine-tuning for 3D object detection, semantic segmentation, and instance segmentation on ScanNet [7], showing the effectiveness of learning 3D features under 4D constraints. Tables 2, 3, and 4 evaluate performance on 3D object detection, semantic segmentation, and instance segmentation, respectively.

Table 2 shows 3D object detection results, for which our pretraining approach improves over baseline training from scratch (+5.5% mAP@0.5) as well as over the strong 3D-based pre-training methods of RandomRooms [31], PointContrast [43] and CSC [17].

In Tables 3 and 4, we evaluate semantic segmentation in comparison with state-of-the-art 3D pre-training approaches [17,43], as well as a baseline training paradigm from scratch. These pre-training approaches improve notably over training from scratch, and our 4DContrast approach leveraging learned representations under 4D constraints, leads to additional performance improvement over train from scratch (+2.3% mIoU for semantic segmentation and +4.2% mAP@0.5 for instance segmentation). We show qualitative results for semantic segmentation in Figure 4.

5.2 SUN RGB-D

We additionally show that our 4DContrast learning scheme can produce transferable representations across datasets. We leverage our pre-trained weights from ScanNet + ModelNet, and explore downstream 3D object detection on the SUN RGB-D [36] dataset. SUN RGB-D is a dataset of RGB-D images, containing 10,335 frames captured with a variety of commodity RGB-D sensors. It contains 3D object bounding box annotations for 10 class categories. We follow the official train/test split of 5,285 train frames and 5,050 test frames.

Table 5 shows 3D object detection performance on SUN RGB-D, with qualitative results visualized in Figure 5. We use the same pre-training as with ScanNet, with downstream fine-tuning on SUN RGB-D data. 4DContrast improves over training from scratch (+6.5% mAP@0.5), with our learned representations surpassing the 3D-based pre-training [17,31,43,46].

Table 3. Semantic segmentation on ScanNet. Our 4D-informed pre-training learns effective features that lead to improved performance boost over training from scratch as well as state-of-the-art 3D-based pre-training of CSC [17] and PointContrast [43].

Method	mIoU	mAcc
Scratch	70.0	78.1
CSC [17]	70.7 (+0.7)	78.8 (+0.7)
PointContrast [43]	71.3 (+1.3)	79.3 (+1.2)
Ours	**72.3** **(+2.3)**	**80.8** **(+2.7)**

Table 4. Instance segmentation on ScanNet. Our 4D-imbued pre-training leads to significantly improved results over training from scratch, as well as favorable performance over other 3D-only pretraining schemes.

Method	mAP@0.5	mIoU
Scratch	53.4	69.0
PointContrast [43]	55.8 (+2.4)	70.9 (+1.9)
CSC [17]	56.5 (+3.1)	71.1 (+2.1)
Ours	**57.6** **(+4.2)**	**71.4** **(+2.4)**

5.3 Data-efficient 3D Scene Understanding

We evaluate our approach in the scenario of limited training data, as shown in Figure 6. 4DContrast improves over baseline training from scratch as well as over state-of-the-art data-efficient scene understanding CSC [17] in semantic segmentation and object detection under various different percentages of ScanNet training data. With only 20% of the training data, we can recover 87% of the fine-tuned semantic segmentation performance training with 100% of the train data from scratch. In object detection, our pre-training enables improved performance for all percentage settings, notably in the very limited regime with +3.0/4.5% mAP@0.5 over CSC/training from scratch at 10% data, and +2.5/5.9% mAP@0.5 with 20% data.

5.4 Ablation Studies

Effect of 3D and 4D Data Augmentation. We consider a baseline variant of our approach that considers only the static 3D scene data without any scene-object augmentations with 3D-3D constraints during pre-training in Table 6 (*Ours (3D data, 3D-3D only)*), which provides some improvement over training from scratch but is notably improved with our 4D pre-training formulation. We additionally consider using our 4D scene-object augmentation with only 3D-3D constraints between sequence frames during pre-training (*Ours (4D data, 3D-3D only)*) in Table 6, which helps to additionally improve performance with implicitly learned priors from 4D data. Both are further improved by our approach to explicitly learn 4D priors in 3D features.

Effect of 4D-invariant Contrastive Priors. In Table 6, we see that learning 4D-invariant contrastive priors through our 3D-4D and 4D-4D constraints during pretraining improves upon data augmentation variants only. Additionally,

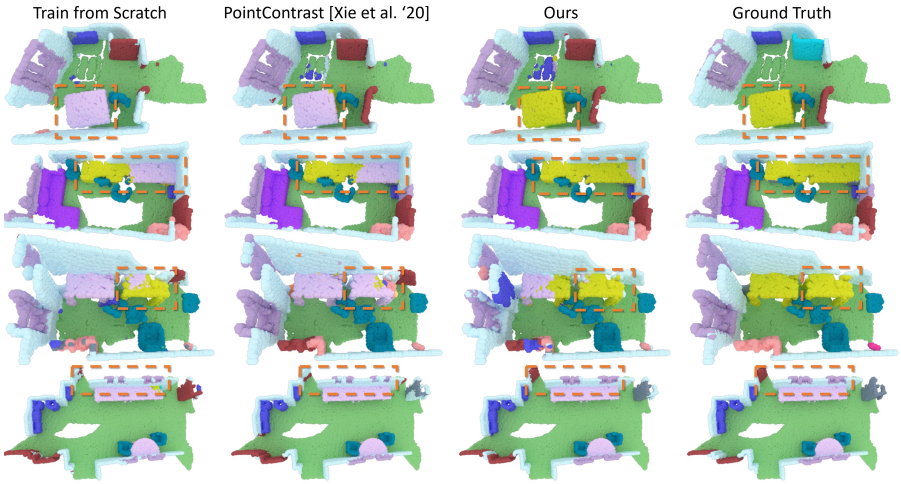


Fig. 4. Qualitative results on ScanNet semantic segmentation. Our 4DContrast pre-training to encode 4D priors enables more consistent segmentation results, in comparison to training from scratch as well as 3D-based PointContrast [43] pre-training.

Table 5. 3D object detection on SUN RGB-D. Our 4D-based pre-training learns effective 3D representations, improving performance over training from scratch and state-of-the-art 3D pre-training methods. *indicates that PointNet++ is used as a backbone instead of a 3D U-Net.

Method	Input	mAP@0.5
VoteNet [27]	Geo + Height	32.9
Scratch + VoteNet [27]	Geo	31.7
PointContrast [43] + VoteNet	Geo	34.8 (+3.1)
RandomRooms [31]* + VoteNet	Geo	35.4 (+3.7)
DeepContrast [46]* + VoteNet	Geo	35.5 (+3.8)
CSC [17] + VoteNet	Geo	36.4 (+4.7)
Ours + VoteNet	Geo	**38.2** (+6.5)

Table 7 evaluates the 3D variant of our approach with our full 4D-based pre-training across a variety of downstream tasks, showing consistent improvements from learned 4D-based priors.

Effect of SimSiam Contrastive Learning. We also consider the effect of our SimSiam contrastive framework as PointContrast [43] leverages a PointInfoNCE contrastive loss. We note that the 3D variant of our approach (*Ours (3D data, 3D-3D only)*) reflects a PointContrast [43] setting using our scene augmentation and SimSiam architecture, which our 4D-based feature learning outperforms.

5.5 Discussion

While 4DContrast pre-training demonstrates the effectiveness of leveraging 4D priors for learned 3D representations, various limitations remain. In particular, 4D feature learning with sparse convolutions involves considerable memory

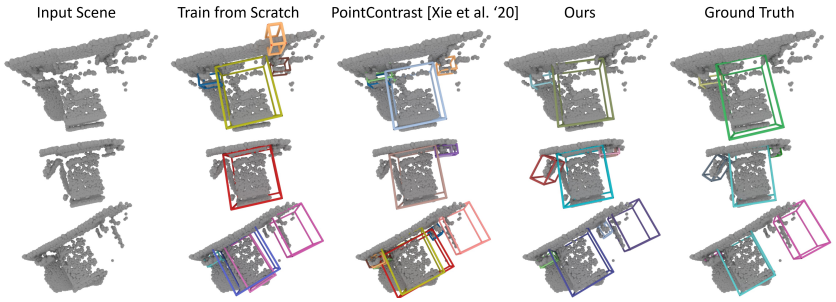


Fig. 5. Qualitative results on SUN RGB-D [36] object detection. Our 4DContrast pre-training to encode 4D priors enables more accurate detection results, in comparison to training from scratch as well as 3D-based PointContrast [43] pre-training. Different colors denote different objects.

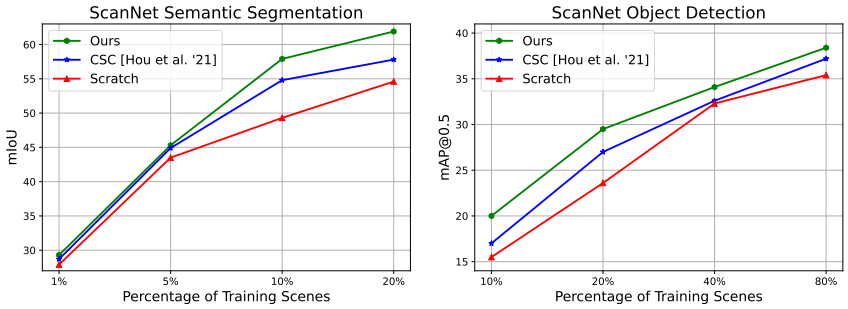


Fig. 6. Data-efficient learning on ScanNet semantic segmentation and object detection. Under limited data scenarios, our 4D-imbued pre-training effectively improves performance over training from scratch as well as the state-of-the-art CSC [17].

during pre-training, so we use half-resolution for characterizing 4D features relative to 3D features and limited sequence durations. Additionally, we consider a subset of 4D motion when augmenting scenes with moving synthetic objects, and believe exploration of articulated motion or more complex dynamic object interactions would lead to additional insight and robustness of learned feature representations.

Memory and Speed. Our 4D-imbued pre-training results in consistent improvements across a variety of tasks and datasets, even with only using the learned 3D backbone for downstream training and inference. Thus, our method maintains the same memory and speed costs for inference as purely 3D-based pre-training approaches. For pre-training, our joint 3D-4D training uses additional parameters (33M for the 4D network in addition to the 38M for the 3D network), but due to jointly learning 4D priors with SimSiam, we do not require as large of a batch size to train as PointContrast [43] (12 vs their 48), nor as many iterations (up to 30K vs 60K), resulting in slightly less total memory use and pre-training time overall.

Table 6. Additionally ablation variants: compared to a baseline of using 3D-3D constraints on static 3D scene data only, leveraging augmented 4D sequence data improves feature learning even under 3D only constraints. Our final 4DContrast pre-training leveraging constraints with learned 4D features achieves the best performance.

Method/Data Augmentation	Pre-training Loss Term(s)	mIoU	mAcc
Scratch	-	70.0	78.1
Ours (3D data, 3D-3D only)	$\mathcal{L}^{3D}$(static scene)	71.5 (+1.5)	79.7 (+1.6)
Ours (4D data, 3D-3D only)	$\mathcal{L}^{3D}$(dynamic scene)	71.7 (+1.7)	80.0 (+1.9)
Ours	$\mathcal{L}^{3D}$,$\mathcal{L}^{3D4D}$,$\mathcal{L}^{4D}$(dynamic scene)	**72.3** (+2.3)	**80.5** (+2.4)

Table 7. Extended ablation of the 3d-only variant of our approach on ScanNet.

Task	Ours (3D data, 3D-3D only)	Ours
Ins. Seg. (mAP@0.5)	54.1	**57.6** (+3.5)
Obj. Det. (mAP@0.5)	38.7	**40.0** (+1.3)
Sem. Seg. 1% data (mIoU)	27.0	**28.2** (+1.2)
Sem. Seg. 5% data (mIoU)	44.4	**45.3** (+0.9)
Sem. Seg. 10% data (mIoU)	56.3	**57.9** (+1.6)
Sem. Seg. 20% data (mIoU)	60.6	**61.9** (+1.3)
Sem. Seg. 100% data (mIoU)	71.5	**72.3** (+0.8)

6 Conclusion

We have presented 4DContrast, a new approach for 3D representation learning that incorporates 4D priors into learned features during pre-training. We propose a data augmentation scheme to construct 4D sequences of moving synthetic objects in static 3D scenes, without requiring any semantic labels. This enables learning from 4D sequences, and we and establish contrastive constraints between learned 3D features and 4D features from the inherent correspondences given in the 4D sequence generation. Our experiments demonstrate that our 4D-imbued pre-training results in performance improvement across a variety of 3D downstream tasks and datasets. Additionally, our learned features effectively transfer to limited training data scenarios, significantly outperforming state of the art in the low training data regime. We hope that this will lead to additional insights in 3D representation learning and new possibilities in 3D scene understanding.

Acknowledgements. This project is funded by the Bavarian State Ministry of Science and the Arts and coordinated by the Bavarian Research Institute for Digital Transformation (bidt), the TUM Institute of Advanced Studies (TUM-IAS), the ERC Starting Grant Scan2CAD (804724), and the German Research Foundation (DFG) Grant Making Machine Learning on Static and Dynamic 3D Data Practical.

References

1. Chang, A., Dai, A., Funkhouser, T., Halber, M., Niebner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from rgb-d data in indoor environments. In: International Conference on 3D Vision. pp. 667–676 (2017) 2, 3
2. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning. pp. 1597–1607 (2020) 3
3. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297 (2020) 3
4. Chen, X., He, K.: Exploring simple siamese representation learning. In: Conference on Computer Vision and Pattern Recognition. pp. 15750–15758 (2021) 4
5. Chen, Y., Liu, J., Ni, B., Wang, H., Yang, J., Liu, N., Li, T., Tian, Q.: Shape self-correction for unsupervised point cloud understanding. In: International Conference on Computer Vision. pp. 8382–8391 (2021) 3
6. Choy, C., Gwak, J., Savarese, S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. In: Conference on Computer Vision and Pattern Recognition. pp. 3075–3084 (2019) 2, 3, 9
7. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Conference on Computer Vision and Pattern Recognition. pp. 5828–5839 (2017) 2, 3, 8, 9, 10
8. Dai, A., Nießner, M.: 3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation. In: European Conference on Computer Vision. pp. 452–468 (2018) 2, 3
9. Engelmann, F., Bokeloh, M., Fathi, A., Leibe, B., Nießner, M.: 3d-mpa: Multi-proposal aggregation for 3d semantic instance segmentation. In: Conference on Computer Vision and Pattern Recognition. pp. 9031–9040 (2020) 3
10. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: Conference on Computer Vision and Pattern Recognition. pp. 3354–3361 (2012) 2, 3
11. Graham, B., Engelcke, M., Van Der Maaten, L.: 3d semantic segmentation with submanifold sparse convolutional networks. In: Conference on Computer Vision and Pattern Recognition. pp. 9224–9232 (2018) 2, 3
12. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P.H., Buchatskaya, E., Doersch, C., Pires, B.A., Guo, Z.D., Azar, M.G., et al.: Bootstrap your own latent: A new approach to self-supervised learning. arXiv preprint arXiv:2006.07733 (2020) 5
13. Han, L., Zheng, T., Xu, L., Fang, L.: Occuseg: Occupancy-aware 3d instance segmentation. In: Conference on Computer Vision and Pattern Recognition. pp. 2940–2949 (2020) 3
14. Hassani, K., Haley, M.: Unsupervised multi-task feature learning on point clouds. In: International Conference on Computer Vision. pp. 8160–8171 (2019) 3
15. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Conference on Computer Vision and Pattern Recognition. pp. 9729–9738 (2020) 3
16. Hou, J., Dai, A., Nießner, M.: 3d-sis: 3d semantic instance segmentation of rgb-d scans. In: Conference on Computer Vision and Pattern Recognition. pp. 4421–4430 (2019) 3, 9
17. Hou, J., Graham, B., Nießner, M., Xie, S.: Exploring data-efficient 3d scene understanding with contrastive scene contexts. In: Conference on Computer Vision and Pattern Recognition. pp. 15587–15597 (2021) 2, 3, 9, 10, 11, 12, 13

18. Hu, W., Zhao, H., Jiang, L., Jia, J., Wong, T.T.: Bidirectional projection network for cross dimension scene understanding. In: Conference on Computer Vision and Pattern Recognition. pp. 14373–14382 (2021) 3
19. Huang, J., Zhang, H., Yi, L., Funkhouser, T., Nießner, M., Guibas, L.J.: Texturenet: Consistent local parametrizations for learning from high-resolution signals on meshes. In: Conference on Computer Vision and Pattern Recognition. pp. 4440–4449 (2019) 2, 3
20. Huang, S., Xie, Y., Zhu, S.C., Zhu, Y.: Spatio-temporal self-supervised representation learning for 3d point clouds. In: International Conference on Computer Vision. pp. 6535–6545 (2021) 2, 3
21. Jiang, L., Zhao, H., Shi, S., Liu, S., Fu, C.W., Jia, J.: Pointgroup: Dual-set point grouping for 3d instance segmentation. In: Conference on Computer Vision and Pattern Recognition. pp. 4867–4876 (2020) 3, 9
22. Kundu, A., Yin, X., Fathi, A., Ross, D., Brewington, B., Funkhouser, T., Pantofaru, C.: Virtual multi-view fusion for 3d semantic segmentation. In: European Conference on Computer Vision. pp. 518–535 (2020) 3
23. Liang, H., Jiang, C., Feng, D., Chen, X., Xu, H., Liang, X., Zhang, W., Li, Z., Van Gool, L.: Exploring geometry-aware contrast and clustering harmonization for self-supervised 3d object detection. In: International Conference on Computer Vision. pp. 3293–3302 (2021) 3
24. Nekrasov, A., Schult, J., Litany, O., Leibe, B., Engelmann, F.: Mix3d: Out-of-context data augmentation for 3d scenes. In: 2021 International Conference on 3D Vision (3DV). pp. 116–125. IEEE (2021) 3
25. Nie, Y., Hou, J., Han, X., Nießner, M.: Rfd-net: Point scene understanding by semantic instance reconstruction. In: Conference on Computer Vision and Pattern Recognition. pp. 4608–4618 (2021) 3
26. Qi, C.R., Chen, X., Litany, O., Guibas, L.J.: Imvotenet: Boosting 3d object detection in point clouds with image votes. In: Conference on Computer Vision and Pattern Recognition. pp. 4404–4413 (2020) 3
27. Qi, C.R., Litany, O., He, K., Guibas, L.J.: Deep hough voting for 3d object detection in point clouds. In: International Conference on Computer Vision. pp. 9277–9286 (2019) 3, 9, 12
28. Qi, C.R., Liu, W., Wu, C., Su, H., Guibas, L.J.: Frustum pointnets for 3d object detection from rgb-d data. In: Conference on Computer Vision and Pattern Recognition. pp. 918–927 (2018) 9
29. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Conference on Computer Vision and Pattern Recognition. pp. 652–660 (2017) 2, 3
30. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: Neural Information Processing Systems (2017) 2, 3
31. Rao, Y., Liu, B., Wei, Y., Lu, J., Hsieh, C.J., Zhou, J.: Randomrooms: Unsupervised pre-training from synthetic shapes and randomized layouts for 3d object detection. In: International Conference on Computer Vision. pp. 3283–3292 (2021) 3, 9, 10, 12
32. Rozenberszki, D., Litany, O., Dai, A.: Language-grounded indoor 3d semantic segmentation in the wild. arXiv preprint arXiv:2204.07761 (2022) 3
33. Sanghi, A.: Info3d: Representation learning on 3d objects using mutual information maximization and contrastive learning. In: European Conference on Computer Vision. pp. 626–642 (2020) 3

34. Sauder, J., Sievers, B.: Self-supervised deep learning on point clouds by reconstructing space. Neural Information Processing Systems (2019) 3
35. Schult, J., Engelmann, F., Kontogianni, T., Leibe, B.: Dualconvmesh-net: Joint geodesic and euclidean convolutions on 3d meshes. In: Conference on Computer Vision and Pattern Recognition. pp. 8612–8622 (2020) 3
36. Song, S., Lichtenberg, S.P., Xiao, J.: Sun rgb-d: A rgb-d scene understanding benchmark suite. In: Conference on Computer Vision and Pattern Recognition. pp. 567–576 (2015) 2, 8, 10, 13
37. Song, S., Xiao, J.: Deep sliding shapes for amodal 3d object detection in rgb-d images. In: Conference on Computer Vision and Pattern Recognition. pp. 808–816 (2016) 9
38. Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E.: Multi-view convolutional neural networks for 3d shape recognition. In: International Conference on Computer Vision. pp. 945–953 (2015) 2
39. Wang, H., Liu, Q., Yue, X., Lasenby, J., Kusner, M.J.: Unsupervised point cloud pre-training via occlusion completion. In: International Conference on Computer Vision. pp. 9782–9792 (2021) 3
40. Wang, P.S., Yang, Y.Q., Zou, Q.F., Wu, Z., Liu, Y., Tong, X.: Unsupervised 3d learning for shape analysis via multiresolution instance discrimination. vol. 35, pp. 2773–2781 (2021) 3
41. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: Conference on Computer Vision and Pattern Recognition. pp. 1912–1920 (2015) 2, 9
42. Xie, Q., Lai, Y.K., Wu, J., Wang, Z., Zhang, Y., Xu, K., Wang, J.: Mlcvnet: Multi-level context votenet for 3d object detection. In: Conference on Computer Vision and Pattern Recognition. pp. 10447–10456 (2020) 3
43. Xie, S., Gu, J., Guo, D., Qi, C.R., Guibas, L., Litany, O.: Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In: European Conference on Computer Vision. pp. 574–591 (2020) 2, 3, 9, 10, 11, 12, 13
44. Yi, L., Zhao, W., Wang, H., Sung, M., Guibas, L.J.: Gspn: Generative shape proposal network for 3d instance segmentation in point cloud. In: Conference on Computer Vision and Pattern Recognition. pp. 3947–3956 (2019) 9
45. Zhang, B., Wonka, P.: Point cloud instance segmentation using probabilistic embeddings. In: Conference on Computer Vision and Pattern Recognition. pp. 8883–8892 (2021) 3
46. Zhang, Z., Girdhar, R., Joulin, A., Misra, I.: Self-supervised pretraining of 3d features on any point-cloud. In: International Conference on Computer Vision. pp. 10252–10263 (2021) 2, 3, 10, 12
47. Zhang, Z., Sun, B., Yang, H., Huang, Q.: H3dnet: 3d object detection using hybrid geometric primitives. In: European Conference on Computer Vision. pp. 311–329 (2020) 3