SODA: Out-of-Distribution Detection in Domain-Shifted Point Clouds via Neighborhood Propagation

Adam Goodge[1], Xun Xu[1] (✉), Bryan Hooi[2], Wee Siong Ng[1], Jingyi Liao[1,2], Yongyi Su[1], and Xulei Yang[1]

[1] Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore
xu_xun@i2r.a-star.edu.sg
[2] School of Computing, National University of Singapore, Singapore

Abstract. As point cloud data increases in prevalence in a variety of applications, the ability to detect out-of-distribution (OOD) point cloud objects becomes critical for ensuring model safety and reliability. However, this problem remains under-explored in existing research. Inspired by success in the image domain, we propose to exploit advances in 3D vision-language models (3D VLMs) for OOD detection in point cloud objects. However, a major challenge is that point cloud datasets used to pre-train 3D VLMs are drastically smaller in size and object diversity than their image-based counterparts. Critically, they often contain exclusively computer-designed synthetic objects. This leads to a substantial domain shift when the model is transferred to practical tasks involving real objects scanned from the physical environment. In this paper, our empirical experiments show that synthetic-to-real domain shift significantly degrades the alignment of point cloud with their associated text embeddings in the 3D VLM latent space, hindering downstream performance. To address this, we propose a novel methodology called SODA which improves the detection of OOD point clouds through a neighborhood-based score propagation scheme. SODA is inference-based, requires no additional model training, and achieves state-of-the-art performance over existing approaches across datasets and problem settings.

Keywords: out-of-distribution detection · point clouds · vision-language models

1 Introduction

Out-of-distribution (OOD) detection is crucial for ensuring the safe and reliable deployment of models in practical applications. It has received significant research attention for 2D images, but remains largely unexplored in the context of 3D point clouds. The detection of unknown 3D objects poses a critical challenge in important applications including robotics, autonomous driving, and healthcare.

Vision-language models (VLMs), such as CLIP [25], have proven effective at OOD detection in image data. OOD inputs are detected by their low cosine similarity to the text embeddings of in-distribution (ID) class labels. Recently, several 3D VLMs incorporating point clouds into this multi-modal latent space have emerged. These models demonstrate strong performance in point cloud classification, but have yet to be evaluated for OOD detection.

A major advantage of VLMs is that they only require ID class labels, not labeled examples of ID data, due to their extensive pre-training on broad data. This is particularly useful in the context of point cloud data, which are typically more expensive to collect and annotate than images. However, point cloud datasets used to pre-train 3D VLMs are also much more limited in size and object diversity than their image counterparts. Moreover, they typically contain only synthetic, computer-designed objects. This results in domain shift when the model is transferred to practical downstream tasks that involve real objects scanned from the physical environment. Figure 1 shows a synthetic chair from ShapeNet [3] and a real chair from ScanObjectNN [28]. Real data can diverge from synthetic data due to several phenomena, such as: non-uniform point density, sensor noise, occlusion, background objects and physical imperfections. As most practical tasks involve real data, it is crucial that models are robust to this domain shift if they are to be deployed in practice.

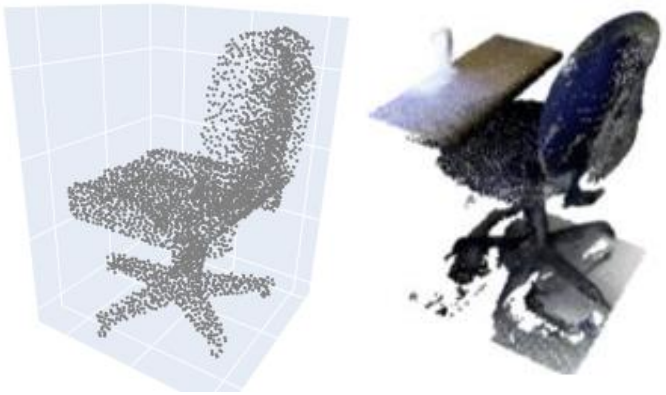

Fig. 1: Example of a synthetic chair sample (left) and real chair sample (right).

In this paper, we show through empirical experiments that "*synthetic-to-real*" domain shift hinders the alignment of the embeddings of real point clouds with their text labels in the latent space of 3D VLMs, evidenced by the degradation in classification performance of real objects. Moreover, we observe that the severity of this degradation has a strong correlation with the severity of the domain shift. However, we also observe that the model is effective at clustering real data by class in the latent space, suggesting that the model maintains an ability to

distinguish between different classes of real objects. These observations motivate our proposed methodology for point cloud OOD detection, called **S**coring for **O**ut-of-Distribution **D**etection through **A**ggregation (**SODA**). We use the similarity of point clouds to ID class labels to initialize OOD scores, followed by an important score refinement step based on a neighborhood-based score propagation scheme which accounts for the severity of domain shift through dynamic 'source-similarity' weighting. We conduct comprehensive experimentation to evaluate our approach and find that it significantly improves OOD detection without additional fine-tuning of the backbone model.

In summary, our contributions are as follows:

- We investigate the effect of synthetic-to-real domain shift on 3D VLMs and find a degradation in the alignment of point cloud and text embeddings.
- We propose SODA, a novel methodology to improve OOD detection in domain shifted point-clouds via domain shift-aware neighborhood propagation.
- We show that this approach achieves state-of-the-art performance through rigorous experiments and ablation study.

The code to reproduce our experiments is available online.

2 Related Work

Numerous methods have been proposed for OOD detection in image data. Notably, MSP [11] uses the maximum softmax probability assigned to a sample by a classification model trained on ID classes, and subsequent methods adopt a similar approach using different confidence measures [14, 19, 20, 27]. However, neural networks are known to make overly confident predictions even for OOD data [10]. OOD point cloud detection remains comparatively under-explored, and existing studies mostly focus on adapting established methods to point-cloud feature extraction models such as PointNet++ [23] and DGCNN [31] in [1], and PointPillars [17] in [13], VAEs [21], and teacher-student models [2]. OpenPatch [24] measures the distance of test sample patches to a memory bank containing patches of ID samples.

Recently, strong performance has been achieved in image-based OOD detection with vision-language models [6, 8, 9, 30]. VLMs are typically trained on massive-scale image datasets that encompass a diverse range of classes, domains, and environments. This extensive exposure enables the model to learn rich visual representations, enhancing its ability to generalize across domains. In the 3D VLM space, two approaches have emerged. One approach is to project point cloud data into image inputs for existing 2D VLMs, examples of which include PointCLIP [37] and CLIP2Point [15]. Another approach is to introduce an additional point cloud encoder and train it to align point cloud embeddings with their paired image and text inputs extracted from a fixed 2D VLM. ULIP [34] and ULIP-2 [35] are notable examples, and the latter achieves state-of-the-art performance in downstream tasks.

As mentioned, point cloud datasets used to pre-train 3D VLMs are relatively limited in size and object diversity, limiting their generalization. Significant research attention has focused on domain adaptation to real point cloud data [16]. Common approaches include domain-invariant training or fine-tuning with task-specific data. In an OOD setting, this means adapting to ID data for each task individually, which is cumbersome or even infeasible in data-scarce scenarios. It also risks worsening performance through catastrophic forgetting [24]. As such, we focus on leveraging the pre-trained embedding space for training-free adaptation to synthetic-to-real domain shift, without adjusting model parameters.

3 Motivation

In this section, we investigate the effect of synthetic-to-real domain shift on the embeddings learnt by 3D VLMs. In our analysis, we use synthetic samples from ModelNet40 [33] and real samples from ScanObjectNN [28]. We focus on only the classes which are common to both datasets (listed under SR1 and SR2 in Table 1) in order to focus on the effect of domain shift and not the semantic differences between classes. We use ULIP-2 [35] as our fixed backbone VLM due to its superior performance. ULIP-2 trains a PointBERT point cloud encoder to align point cloud embeddings with paired image and text caption embeddings learnt by a 2D VLM. We extract the embeddings of both synthetic and real point clouds and make several observations:

Table 1: Class splits for experiments with ScanObjectNN. Classes under SR1 and SR2 overlap with classes of the same name in ModelNet40, but classes under SR3 classes do not.

SR1	SR2	SR3
chair	bed	bag
shelf	toilet	bin
door	desk	box
sink	table	pillow
sofa	display	cabinet

Observation 1: Degraded text-point cloud alignment Following [35], we obtain text embeddings for each class label by creating class-based text prompts from a set of templates (more details in Section 4.2). We classify each sample according to their maximum cosine similarity to these text embeddings. With this setup, we observe that **classification accuracy drops from 94.27% on synthetic data to 73.83% on real data**. This suggests a significant degradation in the alignment of real data with their associated text labels in the latent space compared with synthetic data. This is detrimental to OOD detection, as

OOD data may be inappropriately matched to ID class labels, resulting in false negative errors, and vice versa. However, this degradation is not uniform. We take the average similarity of real samples to their 10 nearest synthetic samples as a measure of how 'close' they are to the source domain. In Figure 2, we see that this 'source similarity' has a direct, linear relationship with classification accuracy. In other words, **real samples that are closer to the source domain are better aligned with the text labels**.

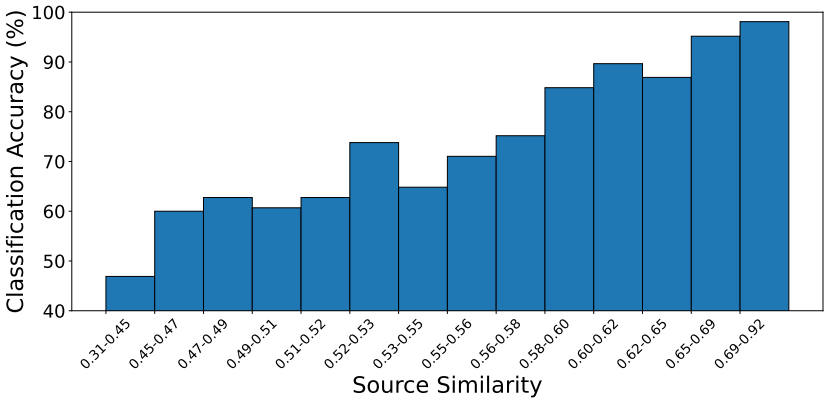

Fig. 2: Classification accuracy (%) of real ScanObjectNN point clouds sorted by cosine similarity to synthetic ModelNet40 samples.

Observation 2: Strong class-based clustering Our second observation is that real samples are strongly clustered by class. Figure 3 visualises the UMAP projections [22] of real point samples, with their colour corresponding to their class label. We see that the vast majority of samples are well clustered according to class. This suggests that, despite a weakened alignment with the text embeddings, the model retains an ability to distinguish between examples of different classes of real objects. This is promising for OOD detection, as it means that we can expect a real sample from an ID class to be mostly neighbored by other samples of the same ID class, and vice versa for OOD samples. These observations provide motivation for our proposed methodology.

4 Methodology

Our work most closely relates to [36], which exploits neighborhood structure within the test set for classification. However, a major difference in OOD detection is that we do not know the label space of OOD samples, which can belong to any unknown class, therefore it is a more challenging open-set problem.

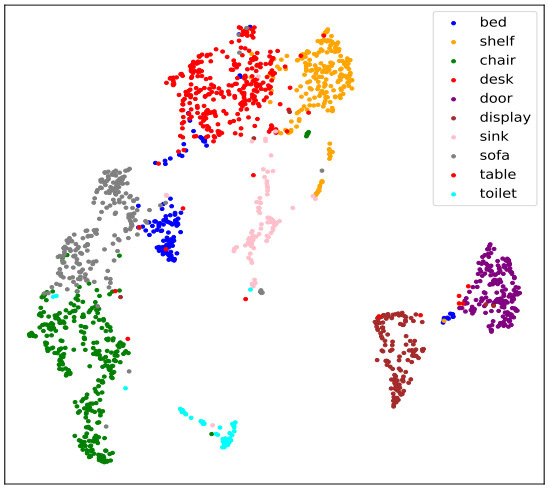

Fig. 3: UMAP projections of real domain test point clouds colored by their class.

Our methodology is inspired by the label propagation algorithm [39], a semi-supervised approach to classification which propagates class information from labeled to unlabeled data. However, as all test data is unlabeled in our setting, we have no ground truth information to propagate. [32] boosts the detection of OOD nodes within a graph by propagating energy-based scores between neighboring nodes. We explore the potential of this framework beyond graph data, particularly for addressing the challenges of domain shift in non-structured data, exploiting the strong class-based latent clustering of 3D VLMs to improve OOD detection performance in a transductive setting. Our methodology is illustrated in Figure 4.

4.1 Problem Statement

We have a set of N ID class labels $\mathcal{C} = \{C_1, C_2, ..., C_N\}$, and an unlabeled test set $\mathbf{X}$, consisting of both ID and OOD real data. A sample is ID if it belongs to any ID class $C_i \in \mathcal{C}$ and OOD otherwise. We aim to distinguish between ID and OOD samples by devising a scoring function which assigns higher scores to ID samples than OOD samples in $\mathbf{X}$.

4.2 SODA

We begin by describing the source-free, zero-shot version of SODA (ZS-SODA), and then explain how source domain samples are incorporated to enhance performance in our full SODA methodology.

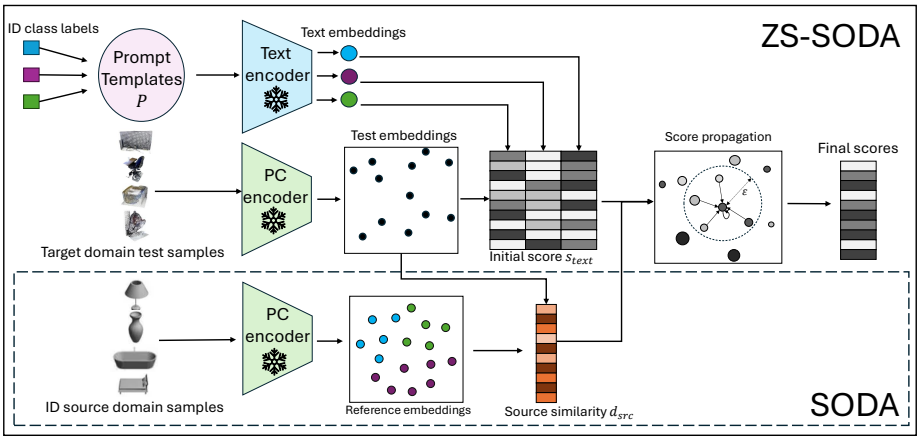

Fig. 4: An overview of SODA. Test samples are assigned initial scores based on their similarity to the ID class text embeddings. If available, these scores are multiplied by their source similarity, and refined via neighborhood propagation within an ε-similarity radius. Target domain images are ScanObjectNN and source domain images are ModelNet40.

Initial Scoring We use a fixed, pre-trained 3D VLM with text encoder g_t and point cloud encoder g_{pc}. Following convention, we use a set of templates $\mathcal{P}$, which are shared between all classes, combined with each class label C_i to create a set of class-specific text prompts $\mathcal{P}_i$. An example template is "*a point cloud model of a CLS*" where *CLS* is replaced with C_i, and the other templates can be found in the supplementary material. These text prompts are encoded and $l2$-normalized, and we take the mean of these embeddings as the text prototype of class C_i:

$$\mathbf{P}_i = \frac{1}{|\mathcal{P}_i|} \sum_{p \in \mathcal{P}_i} g_t(p), \quad i \in 1, .., N. \tag{1}$$

Similarly, we obtain the embedding of a query test sample $\mathbf{x}_i$ via the fixed point cloud encoder: $\mathbf{z}_i = g_{pc}(\mathbf{x}_i)$. With this, we assign the initial score of $\mathbf{x}_i$ as the maximum of its cosine similarities to the text prototypes:

$$s_{text}(\mathbf{x}_i) = \max_{i=1,\dots,N} (sim(\mathbf{z}_i, \mathbf{P}_i)) \tag{2}$$

ID samples are assumed to have higher similarity to their correct class label, resulting in a high score, compared with OOD samples. However, as we observed, the alignment of point cloud embeddings with their text embeddings is degraded by synthetic-to-real domain shift, which challenges this assumption. To address this, we introduce a score refinement step through neighborhood-based score propagation.

Score Propagation The misalignment of real point cloud embeddings with the text embeddings means that the initial scores are prone to substantial un-

certainty or noise. However, we also observed that nearby samples are likely to belong to the same class. Based on this, we refine initial scores using the scores of nearby samples. For example, an ID sample may have an erroneously low score, but by adjusting its score in line with its neighbors, this uncertainty is reduced, the score landscape is smoothened over the latent space, and the overall robustness of OOD scores of individual samples is improved. We achieve this by constructing a *similarity graph*, with a node for each test sample and an edge connecting nodes i and j if their cosine similarity is greater than ε:

$$e_{i,j} = \begin{cases} 1, & \text{if } sim(\mathbf{z_i}, \mathbf{z_j}) \geq \varepsilon \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

We set $\varepsilon = \text{percentile}(\mathbf{S}, 100(1-\eta))$, where $\mathbf{S}$ is the similarities between all pair-wise test samples and η is a hyper-parameter. A smaller η results in a higher ε threshold and a sparser similarity graph, i.e. fewer neighbors per sample on average. This formulation is adaptive to the local density of each sample; densely packed samples will have more neighbors within an ε-similarity radius and therefore more neighbors in the similarity graph, compared to samples in sparse regions. This adaptive strategy is important to limit the propagation of information to only the most similar neighbors, rather than an inflexible k nearest neighbor strategy. We set $\eta = 0.02$ in our main experiments and our ablation study shows that performance is robust within a reasonable range of η.

We update the score of a sample $\mathbf{x}_i$ as follows:

$$s^{(t)}(\mathbf{x}_i) = \alpha s^{(0)}(\mathbf{x}_i) + \frac{1-\alpha}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} s^{(t-1)}(\mathbf{x}_j), \tag{4}$$

where t is the current iteration, $s^{(0)}(\mathbf{x}_i) = s_{text}(\mathbf{x}_i)$ and $\mathcal{N}_i$ is the set of neighbors to node i in the similarity graph. In other words, its updated score is a weighted sum of its initial score and the mean of its neighbors' current scores. Through this procedure, the score of an individual sample is moved closer to those of its neighbors, while anchored to its own initial score to avoid instability, which smoothens the scoring function over the local neighborhood. We allow self-loops in the graph so that every node has at least one neighbor ($|\mathcal{N}_i| > 0 \ \forall i$). We complete T iterations to obtain the final OOD scores. T is a hyper-parameter, and we experiment with different T settings in the ablation study.

4.3 Source Similarity

We observed that real samples that are closer to the source domain in cosine similarity are more likely to be correctly classified through text-based similarity matching. Based on this observation, we hypothesise that these samples are better aligned with the text labels, which means that their text-based OOD scores are more reliable. As such, in our full SODA methodology, we use this source similarity to re-weight the importance of different neighbors during score propagation. More formally, given a set of 'reference samples' from ID classes in

the source domain, $\mathbf{X}^{ref}$, we extract the embeddings of each reference sample $\mathbf{x}_i^{ref}$ from the same fixed point cloud encoder: $\mathbf{z}_i^{ref} = g_{pc}(\mathbf{x}_i^{ref})$. For a test sample $\mathbf{x}_i$, we take the mean of the cosine similarities of $\mathbf{z}_i$ to its top$k$ ($k = 10$) closest reference sample embeddings to define its source similarity, denoted d_{src}:

$$d_{src}(\mathbf{x}_i) = \frac{1}{k} \sum_{j \in \text{top}k(\mathbf{z}_i)} (sim(\mathbf{z}_j^{ref}, \mathbf{z}_i)) \tag{5}$$

To further benefit from neighborhood propagation, we iteratively update d_{src} for each sample according to the same formula as Eq. 4, with $s^{(0)} = d_{src}$. With this, the updated OOD score at the t^{th} iteration is given by:

$$s^{(t)}(\mathbf{x}_i) = d_{src}^{(t)}(\mathbf{x}_i) \cdot s_{text}^{(t)}(\mathbf{x}_i). \tag{6}$$

The effect of this is two-fold. Firstly, the OOD scores of samples with high d_{src} will increase. This is beneficial, as a test sample with greater closeness to source domain ID samples intuitively suggests it is more likely to be ID itself, despite the domain shift. Secondly, this score increase means that this sample has greater influence, or weighting, when its score is propagated to its neighbors in the next iteration, thereby increasing the scores of semantically similar samples in the latent space. After several iterations, score information from reliable ID test samples will flow to their more uncertain ID neighbors, and vice versa for OOD samples, resulting in more robust OOD scores.

5 Experiments

We conduct experiments to study the effectiveness of our methodology in point cloud OOD detection under domain shift. In practical settings, we would assign all test samples with a score below a user-defined threshold as OOD. However, the aim of this work is to improve the OOD scoring process, therefore we use evaluation metrics that do not require the setting of a specific threshold. Namely, AUC (higher is better) and FPR95 (lower is better), which is the false positive rate at 95% recall. We also conduct an ablation study to analyze the behavior of our methodology under different experimental settings.

5.1 Datasets

We follow the experimental framework of [1] in our main experiments, using the most popular benchmark point cloud datasets. In particular, we use real samples from ScanObjectNN [28] as our test data and ModelNet40 [33] as our reference data. We split the object classes into the three subsets shown in Table 1. All of the classes in SR1 and SR2 are also in ModelNet40, whereas SR3 classes are not. As such, we conduct one experiment with SR1 as ID and SR2∪SR3 as OOD classes, and another experiment with SR2 as ID and SR1∪SR3 as OOD classes. The number of reference samples in each class ranges from 109 to 889, and the

Table 2: AUC (higher is better) and FPR95 (lower is better) scores. Best/second best scores are highlighted in bold/underlined. Baselines under '*Customized Model*' are taken from [1] and all train a model using ID source domain data. Baselines under '*Pre-trained Model*' use the pre-trained features from ULIP-2.

	SR1		SR2		Average	
	AUC ↑	FPR95 ↓	AUC ↑	FPR95 ↓	AUC ↑	FPR95 ↓
Customized Model						
MSP [11]	81.0	79.6	70.3	86.7	75.6	83.2
MLS [29]	82.1	76.6	67.6	86.8	74.8	81.7
ODIN [19]	81.7	77.3	70.2	84.4	76.0	80.8
Energy [20]	81.9	77.5	67.7	87.3	74.8	82.4
GradNorm [14]	77.6	80.1	68.4	86.3	73.0	83.2
ReAct [27]	81.7	75.6	67.6	87.2	74.6	81.4
NF [1]	78.0	84.4	74.7	84.2	76.4	84.3
OE+mixup [12]	71.2	89.7	60.3	93.5	65.7	91.6
ARPL+CS [4]	82.8	74.9	68.0	89.3	75.4	82.1
Cosine proto [7]	79.9	74.5	76.5	77.8	78.2	76.1
CE (L2) [1]	79.7	84.5	75.7	80.2	77.7	82.3
SubArcFace [5]	78.7	84.3	75.1	83.4	76.9	83.8
Pre-trained Model						
MSP* [11]	83.0	84.2	74.6	81.4	78.8	82.8
MLS* [29]	81.0	79.4	83.2	62.7	82.1	71.0
Cosine Proto [7]	80.7	70.2	73.6	83.3	77.1	76.8
Mahalanobis [18]	73.8	89.7	65.3	83.3	69.5	86.5
OpenPatch [24]	85.8	<u>54.4</u>	71.6	74.1	78.7	64.3
ZS-SODA* (*Ours*)	<u>85.9</u>	67.1	<u>87.1</u>	<u>50.4</u>	<u>86.5</u>	<u>58.7</u>
SODA (*Ours*)	**93.3**	**33.3**	**87.7**	**47.4**	**90.5**	**40.4**

*source-free methods

details can be found in the supplementary material. Following [1], we randomly sample 1024 points from reference point clouds and 2048 points from test point clouds.

We also experiment with ModelNet-C [26], which contains corrupted versions of ModelNet40 data, with multiple types of corruptions which are commonly observed in real data, such as global and local noise and point dropout. In these experiments, we use the clean ModelNet40 ID samples as reference samples and the corrupted ModelNet-C samples as test samples. We use the strongest level of corruption in our experiments, as this represents the greatest degree of domain shift. We conduct two types of experiments, one with each of SR1 and SR2 from Table 1 as the ID/OOD samples respectively.

Table 3: AUC (higher is better) and FPR95 (lower is better) performance of all pre-trained methods for ScanObjectNN test samples. ZS-SODA and SODA without propagation are simply the initial scores. Avg. change shows the average change after propagation over all methods.

	S1		S2		Average	
	AUC ↑	FPR95 ↓	AUC ↑	FPR95 ↓	AUC ↑	FPR95 ↓
Without Propagation						
MSP	83.0	84.2	74.7	81.4	78.8	82.8
Source Similarity	86.6	58.5	79.2	71.1	82.9	64.8
Cosine Proto	80.7	70.2	73.6	83.3	77.1	76.8
ZS-SODA	81.0	79.4	83.2	62.7	82.1	71.0
SODA	88.6	63.0	84.8	58.0	86.7	60.5
With Propagation						
MSP	87.7	61.9	78.7	71.3	83.2	66.6
Source Similarity	90.1	43.2	84.5	55.2	87.3	49.2
Cosine Proto	82.5	56.8	74.2	76.2	78.3	66.5
ZS-SODA	85.9	67.1	87.1	50.4	86.5	58.7
SODA	**93.3**	**33.3**	**87.7**	**47.4**	**90.5**	**40.4**
Avg. change	4.0	-19.4	2.9	-10.0	3.5	-14.7

5.2 Baselines

Under '*Customized Model*', we adopt the methods presented in [1] as baselines. All of these methods use models that have been trained on exclusively ID data from the source domain. We present their PointNet++ results due to its superior performance. Under, '*Pre-trained Model*', we adopt methods that use pre-trained models. OpenPatch [24] and Mahalanobis [18], which measures the Mahanalobis distance to the reference samples, use a single modality PointBERT backbone. We also implement MSP, MLS and Cosine Proto, which measures the maximum similarity to ID reference sample prototypes, using the same features extracted from the 3D VLM ULIP-2 (PointBERT backbone) as ZS-SODA and SODA. For hyper-parameters, we use $T = 5$, $\alpha = 0.2$ and $\eta = 0.02$ by default without tuning. The effect of hyper-parameter settings is shown in the ablation study. We run our experiments in PyTorch on an Nvidia A5000 24G GPU.

5.3 Results

ScanObjectNN Table 2 shows the average AUC and FPR95 scores over three random trials, for ID classes SR1 and SR2 separately, as well as their average. The standard deviations are shown in the supplementary material. We see that pre-trained methods mostly out-perform the customized model methods. This can be explained by the difference in model architecture as well as the superior representation learning that results from larger scale pre-training. We also see

that our methodology significantly improves performance over all baselines. We see that ZS-SODA and SODA achieve an average of 3.9 and 8.5 percentage points improvement in AUC score over the next best-performing baseline, and a -13.4 and -31.4 point reduction in FPR95 respectively. SODA is also computationally efficient; we show in the supplementary material that similarity graph construction and score propagation contribute only a tiny portion of the overall runtime compared to the feature extraction phase. In the supplementary material, we also conduct experiments using both reference and test samples from ShapeNet [3] (i.e. all synthetic) and both reference and test samples from ScanObjectNN (i.e. all real), and find a similar improvement from our methodology.

We also measure the impact of score propagation using the other OOD methods as the initial scores, namely MSP, Cosine Proto and the source similarity d_{src} itself. Table 3 shows that score propagation greatly improves performance, with an average improvement of 3.5 points in AUC and -14.7 points in FPR95 across all methods, which demonstrates the positive effect of score propagation on OOD detection. We also see that SODA consistently outperforms the other methods, both with and without propagation, which demonstrates the complementary benefits of accounting for the similarity to the source domain alongside text similarity in detecting domain-shifted OOD inputs.

Table 4: Performance of ZS-SODA and SODA (and change in performance from inital scores before propagation) on corrupted ModelNet-C test samples.

	ZS-SODA		SODA	
Corruption	AUC ↑	FPR95 ↓	AUC ↑	FPR95 ↓
Add Global	89.7 (+2.7)	41.4 (-11.5)	97.1 (+1.5)	20.9 (-9.0)
Add Local	90.0 (+5.5)	52.4 (-11.8)	95.7 (+3.9)	23.3 (-26.9)
Dropout Global	76.9 (+3.9)	75.3 (-7.9)	88.4 (+5.7)	62.3 (-12.7)
Dropout Local	76.6 (+3.2)	74.9 (-6.6)	86.7 (+3.5)	58.9 (-11.8)
Jitter	48.6 (-0.6)	97.8 (+1.6)	60.5 (+1.5)	88.0 (-2.7)
Rotate	86.2 (+3.1)	55.5 (-6.8)	95.4 (+2.7)	35.9 (-8.4)
Scale	87.6 (+1.8)	35.4 (-13.2)	96.5 (+1.1)	23.5 (-0.4)
Average	**79.4 (+2.8)**	**61.8 (-8.0)**	**88.6 (+2.8)**	**44.7 (-10.3)**

ModelNet-C Table 4 shows the average performance with ModelNet-C test samples. For brevity, we show the average results over both experimental setups for ZS-SODA and SODA, followed by the change in these metrics from the initial scores before score propagation (in parentheses). We see that our methodology consistently improves performance in both metrics across corruption types, except for ZS-SODA which declines slightly for the jitter corruption. On average, we see a 2.8 point improvement in AUC for both ZS-SODA and SODA, and a

$-8.0/-10.3$ improvement in FPR95 scores for ZS-SODA/SODA. The full results are shown in the supplementary.

5.4 Ablation Study

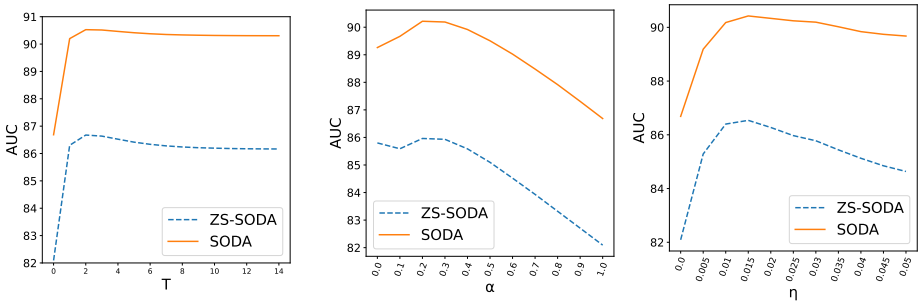

Fig. 5: Average AUC performance with different numbers of iterations, T (left), settings of α (middle), and settings of η (right).

Hyper-parameters Figure 5 shows how performance varies with different settings of hyper-parameters. Firstly, the leftmost figure shows AUC performance after different numbers of propagation iterations. $T = 0$, gives the initial score, from which we see a large increase in performance after just one iteration. Performance slightly drops before converging at around $T = 6$ iterations, after which the OOD scores are fully converged and there is no more change in performance. In the middle figure, we see how performance changes as α in Eq. 4 varies. $\alpha = 1$ is equivalent to the initial scores, while a smaller α gives greater weight to the current score information from neighbors in score updates. We see that optimal performance is achieved around $\alpha \approx 0.25$, which signifies the importance of neighborhood information in refining the initial scores. The rightmost figure shows performance for different settings of η. A larger η corresponds to a lower ε threshold in Eq. 3, and consequently a denser similarity graph with more connecting edges between samples. We see peak performance is achieved at $\eta \approx 0.015$, and slowly declines from there. As η increases, there are more samples propagating score information to neighbors of lower cosine similarity, which eventually hinders performance as the propagated information becomes less relevant. In practice, the setting of these hyper-parameters are guided by the number of available samples and also the sensitivity of individual applications to the trade-off between false positive and negative errors.

Backbone models Table 5 shows the performance of SODA before and after score propagation using different backbone 3D VLMs: ULIP-2, ULIP [34] and

Table 5: AUC and FPR95 performance of SODA without propagation (the initial score) and with propagation for different backbone feature extraction models.

	Without Propagation		With Propagation		Avg. change	
	AUC ↑	FPR95 ↓	AUC ↑	FPR95 ↓	AUC ↑	FPR95 ↓
PointCLIPv2	63.7	91.4	65.3	86.5	1.6	-4.9
ULIP	78.5	76.7	82.4	64.4	3.9	-12.3
ULIP-2	**86.7**	**60.5**	**90.5**	**40.4**	**3.8**	**-20.1**

PointCLIPv2 [38]. We see that score propagation consistently improves both AUC and FPR95 metrics across all backbone models. Furthermore, we see that this improvement is most pronounced with ULIP-2, which also achieves the best performance overall. This aligns with its superior performance in other downstream tasks demonstrated in the original work. We can expect a better model to learn more meaningful latent representations, resulting in more semantically meaningful neighbors which are beneficial for score propagation.

In the supplementary material, we perform several additional experiments. Firstly, we test different formulations of text prompt templates and find that different prompts give better performance in different experimental setups. Overall, using the average of the embeddings from multiple prompt templates is more robust than any single template.

6 Conclusion

We investigate the effect of synthetic-to-real domain shift on the embeddings learnt by pre-trained 3D vision-language models and find that the alignment of real domain point clouds with their corresponding text labels is degraded compared with synthetic data. This has significant implications for OOD detection and other practical downstream tasks that concern real point cloud data. To address this, we propose a novel methodology called SODA which updates and refines the OOD scores assigned to test samples using score information propagated from similar, nearby points within their local neighborhood. This has a local smoothening effect on the scoring function which improves robustness and significantly enhances detection performance. We achieve state-of-the-art performance in both AUC and FPR95 metrics across different experimental settings.

Acknowledgments. This research work is supported by the Agency for Science, Technology and Research (A*STAR) under its MTC Programmatic Funds (Grant No. M23L7b0021).

References

1. Alliegro, A., Cappio Borlino, F., Tommasi, T.: 3dos: Towards 3d open set learning-benchmarking and understanding semantic novelty detection on point clouds. NeurIPS **35**, 21228–21240 (2022)

2. Bhardwaj, A., Pimpale, S., Kumar, S., Banerjee, B.: Empowering knowledge distillation via open set recognition for robust 3d point cloud classification. Pattern Recognition Letters **151**, 172–179 (2021)
3. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)
4. Chen, G., Peng, P., Wang, X., Tian, Y.: Adversarial reciprocal points learning for open set recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(11), 8065–8081 (2021)
5. Deng, J., Guo, J., Liu, T., Gong, M., Zafeiriou, S.: Sub-center arcface: Boosting face recognition by large-scale noisy web faces. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16. pp. 741–757. Springer (2020)
6. Esmaeilpour, S., Liu, B., Robertson, E., Shu, L.: Zero-shot out-of-distribution detection based on the pre-trained model clip. In: AAAI. vol. 36, pp. 6568–6576 (2022)
7. Fontanel, D., Cermelli, F., Mancini, M., Caputo, B.: Detecting anomalies in semantic segmentation with prototypes. In: CVPR. pp. 113–121 (2021)
8. Fort, S., Ren, J., Lakshminarayanan, B.: Exploring the limits of out-of-distribution detection. NeurIPS **34**, 7068–7081 (2021)
9. Goodge, A., Hooi, B., Ng, W.S.: When text and images don't mix: Bias-correcting language-image similarity scores for anomaly detection. arXiv preprint arXiv:2407.17083 (2024)
10. Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural networks. In: International conference on machine learning. pp. 1321–1330. PMLR (2017)
11. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. arXiv preprint arXiv:1610.02136 (2016)
12. Hendrycks, D., Mazeika, M., Dietterich, T.: Deep anomaly detection with outlier exposure. arXiv preprint arXiv:1812.04606 (2018)
13. Huang, C., Abdelzad, V., Mannes, C.G., Rowe, L., Therien, B., Salay, R., Czarnecki, K., et al.: Out-of-distribution detection for lidar-based 3d object detection. In: ITSC. pp. 4265–4271. IEEE (2022)
14. Huang, R., Geng, A., Li, Y.: On the importance of gradients for detecting distributional shifts in the wild. NeurIPS **34**, 677–689 (2021)
15. Huang, T., Dong, B., Yang, Y., Huang, X., Lau, R.W., Ouyang, W., Zuo, W.: Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In: ICCV. pp. 22157–22167 (2023)
16. Huch, S., Lienkamp, M.: Towards minimizing the lidar sim-to-real domain shift: Object-level local domain adaptation for 3d point clouds of autonomous vehicles. Sensors **23**(24), 9913 (2023)
17. Lang, A.H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O.: Pointpillars: Fast encoders for object detection from point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12697–12705 (2019)
18. Lee, K., Lee, K., Lee, H., Shin, J.: A simple unified framework for detecting out-of-distribution samples and adversarial attacks. NeurIPS **31** (2018)
19. Liang, S., Li, Y., Srikant, R.: Enhancing the reliability of out-of-distribution image detection in neural networks. arXiv preprint arXiv:1706.02690 (2017)
20. Liu, W., Wang, X., Owens, J., Li, Y.: Energy-based out-of-distribution detection. NeurIPS **33**, 21464–21475 (2020)

21. Masuda, M., Hachiuma, R., Fujii, R., Saito, H., Sekikawa, Y.: Toward unsupervised 3d point cloud anomaly detection using variational autoencoder. In: ICIP. pp. 3118–3122. IEEE (2021)
22. McInnes, L., Healy, J., Melville, J.: Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426 (2018)
23. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. NeurIPS **30** (2017)
24. Rabino, P., Alliegro, A., Borlino, F.C., Tommasi, T.: Openpatch: a 3d patchwork for out-of-distribution detection. arXiv preprint arXiv:2310.03388 (2023)
25. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML. pp. 8748–8763. PMLR (2021)
26. Ren, J., Pan, L., Liu, Z.: Benchmarking and analyzing point cloud classification under corruptions. arXiv:2202.03377 (2022)
27. Sun, Y., Guo, C., Li, Y.: React: Out-of-distribution detection with rectified activations. NeurIPS **34**, 144–157 (2021)
28. Uy, M.A., Pham, Q.H., Hua, B.S., Nguyen, D.T., Yeung, S.K.: Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In: ICCV (2019)
29. Vaze, S., Han, K., Vedaldi, A., Zisserman, A.: Open-set recognition: A good closed-set classifier is all you need? arXiv preprint arXiv:2110.06207 (2021)
30. Wang, H., Li, Y., Yao, H., Li, X.: Clipn for zero-shot ood detection: Teaching clip to say no. In: ICCV. pp. 1802–1812 (2023)
31. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph cnn for learning on point clouds. ACM Transactions on Graphics **38**(5), 1–12 (2019)
32. Wu, Q., Chen, Y., Yang, C., Yan, J.: Energy-based out-of-distribution detection for graph neural networks. arXiv preprint arXiv:2302.02914 (2023)
33. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: CVPR. pp. 1912–1920 (2015)
34. Xue, L., Gao, M., Xing, C., Martín-Martín, R., Wu, J., Xiong, C., Xu, R., Niebles, J.C., Savarese, S.: Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding. In: CVPR. pp. 1179–1189 (2023)
35. Xue, L., Yu, N., Zhang, S., Li, J., Martín-Martín, R., Wu, J., Xiong, C., Xu, R., Niebles, J.C., Savarese, S.: Ulip-2: Towards scalable multimodal pre-training for 3d understanding. arXiv preprint arXiv:2305.08275 (2023)
36. Yang, S., Van de Weijer, J., Herranz, L., Jui, S., et al.: Exploiting the intrinsic neighborhood structure for source-free domain adaptation. Advances in neural information processing systems **34**, 29393–29405 (2021)
37. Zhang, R., Guo, Z., Zhang, W., Li, K., Miao, X., Cui, B., Qiao, Y., Gao, P., Li, H.: Pointclip: Point cloud understanding by clip. In: CVPR. pp. 8552–8562 (2022)
38. Zhu, X., Zhang, R., He, B., Guo, Z., Zeng, Z., Qin, Z., Zhang, S., Gao, P.: Pointclip v2: Prompting clip and gpt for powerful 3d open-world learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2639–2650 (2023)
39. Zhu, X., Ghahramani, Z., Lafferty, J.D.: Semi-supervised learning using gaussian fields and harmonic functions. In: (ICML-03). pp. 912–919 (2003)