This WACV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

DisCo: Discovering Common Affordance from Large Models for Actionable Part Perception

Youpeng Wen[1*], Yi Zhu[2*], Zhihao Zhan[1], Pengzhen Ren[3], Jianhua Han[2],
Hang Xu[2], Shen Zhao[1†], Xiaodan Liang[13†]
[1]Shenzhen Campus of Sun Yat-Sen University,
[2]Huawei Noah's Ark Lab, [3]Peng Cheng Laboratory

Abstract

*Actionable part perception for robotic object manipulation needs to perceive parts over open-world object categories within 3D space, which is challenging as the appearance of the same part on different objects varies greatly. It is frequently observed that despite the huge intra-class difference in appearance, the parts share common interactive functions over different objects, i.e., common affordance. According to this observation, we propose **DisCo**, a novel technique that **Dis**covers **Co**mmon affordance information from powerful large models for guiding the actionable part perception across open-world objects. Specifically, we first use a large language model to identify the object names that each part potentially belongs to and a text-to-image generative model to generate image examples for the queried objects, constructing image-text paired data that indicate visual and semantic information of common affordance. Then, our model encodes the common affordance information by learning to pair the object-part images with their text descriptions. Subsequently, the 2D-pixel features are distilled into 3D space, thus the 3D point features are enriched with not only the semantic information of open-set objects but also the common affordance information which is highly generalizable. Finally, a segmentation head and a pose regression network are developed to predict more accurate results of part segmentation and pose estimation, improving the success rate of robotic object manipulation. Extensive experiments show that our method outperforms existing methods on the part instance and semantic segmentation by significant margins of 4.8% mAP, 5.4% AP50, and 3.9% mIoU on the unseen object categories.*

1. Introduction

Being able to identify actionable parts across various object categories [8, 12, 25, 32, 33] is crucial for versatile object manipulation—a vital skill necessary for Embodied AI agents interacting within dynamic real-world environments.

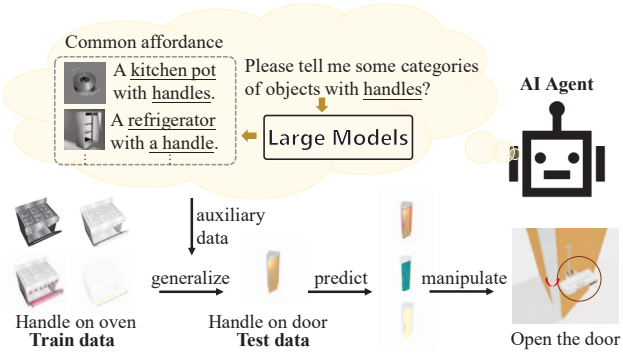


Figure 1. We propose to **discover common affordance**, which corresponds to the highly generalizable cues for object-part interaction across open-world object classes, using large generative and discriminative models.

Current research approaches [7, 8, 19] pose actionable part perception as a robust 3D segmentation challenge. In this paradigm, objects are treated as distinct domains, and models are trained to learn domain-invariant representations of parts through adversarial training methods. However, these learned representations often face challenges dealing with significant visual variations among the same part category across different objects. Moreover, they commonly miss capturing the subtle visual distinctions that differentiate one part class from another. Another viable solution involves directly applying existing open-world segmentation methods to segment actionable parts. However, these approaches usually detect all visible object parts in an uncontrolled environment. Yet, actionable part segmentation focuses on specifically identifying object parts that can be interacted with, such as handles, buttons, or switches. It focuses on recognizing the functional aspects of objects that enable interaction. Therefore, our objective is to understand these functional aspects of objects and enhance actionable part perception.

*Equal contribution, †Corresponding author

2642-9381/25/$31.00 ©2025 IEEE

Common affordance, which refers to the shared interactive functions within the same part category across various objects, can be uncovered by analyzing ample object examples containing these parts. These examples often provide clear indications of how objects and their parts interact. For instance, both doors and microwave ovens commonly have "handles" that can be opened. Motivated by this observation, we explore the broader and more adaptable common affordance information present in objects. This exploration aims to grasp the functional aspects of objects.

Building upon recent advancements in generative and discriminative large-scale models, this paper proposes **DisCo**, which **Dis**covers **Co**mmon affordance from these large models, thereby improving actionable part perception. The large models consist of a large language model associating each part category with potential open-world object labels, a generative text-to-image diffusion model crafting object-part images that present common affordance, and a cross-modal pretrained model offering knowledge of fine-grained visual-semantic alignment for part segmentation across diverse open-world objects. They serve as abundant sources of visual intricacies and potential interactions among object parts, conveniently available off the shelf, as shown in Figure 1.

Specifically, the **DisCo** model is developed to **query**, **learn**, and **distill** common affordance for developing more robust and distinct representations for actionable part perception. First, we **query** an image-text paired dataset as auxiliary data from a Large Language Model (LLM) and a diffusion model. This dataset provides comprehensive insights into both the visual aspects and interactive functionalities of object parts, demonstrating common affordance across diverse objects in the open world. Subsequently, DisCo **learns** to encode this common affordance information into fine-grained image features by ensuring consistency between the embeddings of generated images and their corresponding object-part texts (e.g., "a door with a handle"). This process achieves a consistent and generalizable alignment between 2D visual information and textual descriptions for objects and actionable parts. Following this, DisCo **distills** the acquired common affordance knowledge, initially learned at the 2D pixel level, into their 3D point feature equivalents. This enriches the 3D point features, enhancing their capacity for effective generalization. Finally, these refined 3D features are utilized to predict part segmentation and estimate poses, facilitating robotic object manipulation.

To validate the generalization ability of DisCo, we conduct extensive experiments on the GAPartNet dataset [8] to show that our method achieves significant improvements over existing state-of-the-art on unseen objects. We also assess the estimated object pose for manipulation within simulated environments, successfully executing precise control operations even with exceedingly small parts.

The main contributions in this paper are three-fold:

- We address the challenging problem of identifying actionable parts by extracting common affordance from extensive generative and discriminative models, then distilling this knowledge to aid in segmenting parts within a 3D space.

- The common affordance querying, learning, and distilling modules work in tandem to seamlessly integrate both the visual appearance and the functional interaction information of actionable parts and their respective objects, which notably boosts the generalization ability of object manipulation.

- Our method is demonstrated to be superior over existing state-of-the-art (improving about 5% on almost all the metrics over unseen object classes), which greatly facilitates object manipulation.

2. Related Work

3D Part Instance Segmentation. General point cloud segmentation has been extensively studied, employing either point-based [21, 22, 36] or voxel-based [10, 38] methods due to variant data inputs, with pioneering methods like PointNet [21] for point-by-point feature learning in unstructured point clouds and VoxelNet [38] for voxel rasterization and convolutional processing of 4D tensors. In the robot operation scenario, the segmentation of actionable object parts has drawn great attention. Large-scale datasets [2, 16, 18, 35] of 3D shapes with annotations of actionable parts form the fundamental for the task, *e.g.*, AKB-48 [16] and PartNet [18]. Several advancements [8, 17] have been achieved in the development of part instance segmentation. The most comparable method, GAPartNet [8] uses adversarial strategies to compensate for cross-domain degradation but will cause instability in training. In contrast to the methods mentioned above, we tackle the problem of actionable part segmentation from a brand-new perspective. We propose to make full use of off-the-shelf large models and collect common affordance information for the parts.

Open-World 3D Segmentation. 3D segmentation in the open world aims to segment and identify untrained 3D point cloud objects, vital for applications like autonomous systems and robotics where novel or unexpected objects are frequently encountered. Some open-world segmentation models [5, 15, 20, 26] utilize techniques like zero-shot or few-shot learning to adapt to new objects, with notable developments like PLA's [5] language-aware embeddings model and OpenMask3D's [26] open-vocabulary method for 3D instance segmentation. Another branch of work uses large-scale, diverse datasets and synthetic data generation to train these models. These approaches [24, 37] introduce a wide range of object types and scenarios, enhancing the robustness and versatility of the segmentation models. We have

integrated the two main methods mentioned above, leveraging both pretrained models and synthetic data to assist in the open-world 3D part segmentation task.

Object Pose Estimation and Manipulation. Object pose estimation and manipulation primarily revolve allowing robots to accurately detect, identify, position, grip, and manipulate objects. Notable studies include NOCS [30], FS-Net [3], CASS [31], 6-PACK [29], and Captra [34], which feature unique approaches such as normalized object coordinate space and canonical latent shape space. Few studies address moving actionable parts of objects, though recent ones like Gadre et al. [6], leverage low-level generalizability. However, these methods only deal with simple objects, highlighting the need for better geometric and actionable pattern consistency in more complex objects. In our task, we focus on the manipulation of actionable parts of open-world objects.

3. Method

3.1. Preliminaries

Problem Definition. Given the 3D point cloud observation of the object $\mathbf{P} \in N \times 3$, where N denotes the number of points, the agent is first required to segment out the actionable parts on the presently observed object and estimate the part's pose for subsequent object manipulation. In real-world scenarios, actionable parts across diverse objects often exhibit similar geometries and functionalities. The actionable part perception network is trained using a dataset comprising known objects (e.g., "handle" of a "door") and aims to generalize effectively to unseen objects (e.g., "handle" of a "teapot"). The GAPart [8] dataset categorizes these actionable parts into 9 distinct classes based on their shape and degree of operational freedom. For each actionable part i, the output of the perception network includes a segmentation mask M_i, a class label $l_i \in \{1, ..., 9\}$, and a 7-dimensional rigid transformation of the pose.

Learning Framework. The primary challenge in actionable part perception lies in its ability to generalize across open-world objects in real-world environments. Current methods train perception models on a restricted set of source objects, leaving them susceptible to dataset biases and limiting their generalization capabilities. To overcome this constraint, we propose leveraging the scalability advantage of 2D image perception to enhance 3D actionable object part perception. This enhancement involves querying, learning, and distilling extensive knowledge of common affordance from large-scale pretrained models. In Figure 2, the common affordance querying module initially gathers visual and textual insights about object-part interactions by leveraging a Large Language Model (LLM) and a Diffusion Model (DM). Subsequently, the common affordance learning and distillation module incorporates this queried knowledge into perception model training, thereby bolstering part perception capabilities. We employ Lseg's [14] pretrained ViT-L model as our image encoder E^{2D}, CLIP's [23] pretrained text transformer as our text encoder E^{text}, and a Sparse UNet [9] as our 3D backbone E^{3D} which extracts 3D features from point cloud. To maintain 2D-3D consistency, we use a domain adaptor and a semantic adaptor to project 2D features and 3D features to the same dimension. Then we leverage 2D-3D alignment and 3D-text alignment to learn generalization ability from the pretrained 2D models and get semantic label for each voxel. For learning common affordance from generated images, we use contrastive loss to align 2D features and text features. Next, we use the grouping and scoring module to obtain instance segmentation predictions. After that, the point-wise pose is estimated through the pose estimation network, and the part can be manipulated through the predicted pose and segmentation results.

3.2. Common Affordance Querying

We query common affordance by gathering an image-text paired dataset from large models. Given an actionable part category $l^p \in \mathcal{C}^p$, we feed l^p prompted with T^p = "Please tell me some categories of objects with {part name}" into large language model (LLM) to get the object labels $\{l^o_1, ..., l^o_{\hat{C}_p}\}$ that potentially contain the part l^p in daily life, as:

$$\{l^o_1, ..., l^o_{\hat{C}_p}\} = LLM(T^p(l^p)), \tag{1}$$

where $\hat{C}_p$ is the number of the queried object names. For example, we input "Please tell me some categories of objects with handles" as the prompt to ChatGPT to get "luggage, teapot, ..." that potentially have "handles".

Then we prompt both the part and their queried object names with T^{po} = "a {*object name*} with a {part name}" and fed the texts into a Diffusion Model to generate RGB images $\{I^o_1, ..., I^o_{\hat{C}_p}\}$ that show the objects, as:

$$\{I^o_1, ..., I^o_{\hat{C}_p}\} = DM(T^{po}(\{l^o_1, ..., l^o_{\hat{C}_p}\})). \tag{2}$$

Based on the generated image-text pairs we can build an auxiliary dataset $\mathcal{A} = \{\{I_1; T_1\}, ..., \{I_M; T_M\}\}$ that indicates the common affordance information of actionable part categories. M is the number of pairs.

3.3. Common Affordance Learning

Our model learns to encode common affordance information via contrasting the image-text pairs from the auxiliary dataset $\mathcal{A}$. Specifically, the images are fed into the image encoder E^{2D} (e.g., a pretrained ViT-L from LSeg [14]) followed with one MLP layer as Domain Adaptor ϕ^{DA} to adapt the pretrained features to the domain of actionable part perception, and one pooling layer to aggregate the fine-grained image features $\mathbf{F}^{2D} \in \mathbb{R}^{H \times W \times d}$ which will be used in our distillation into image-level embeddings

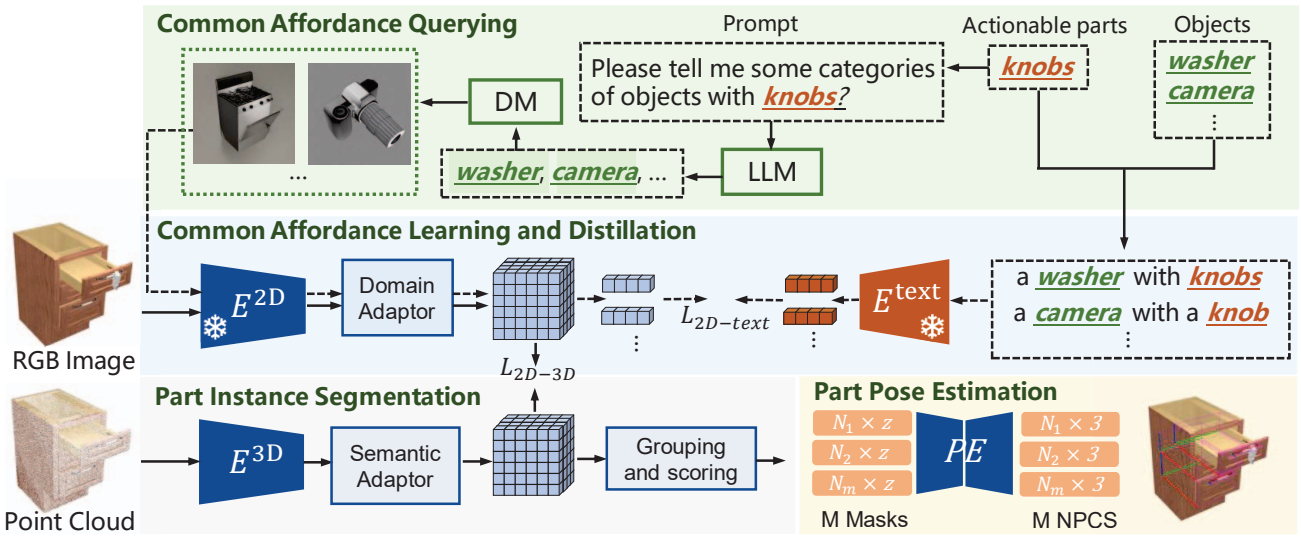


Figure 2. **Overall architecture**. DisCo begins by querying common affordance from large models and assembling an image-text dataset that showcases object-part interactions. Using this dataset, the model learns to encode both visual and semantic common affordance details, yielding robust and refined features in the 2D domain. Subsequently, our model distills these 2D features into their 3D equivalents, transferring crucial common affordance information to enhance actionable part perception. Finally, our framework predicts part semantic and instance segmentation masks, along with pose estimation results, enabling generalized object manipulation in real-world scenarios.

$\mathbf{V} = \{\mathbf{v}_i \in \mathbb{R}^d\}_{i=1}^M$. Meanwhile, for each image the text prompt encoded via E^{text}, resulting in text embeddings $\mathbf{T} = \{\mathbf{t}_i \in \mathbb{R}^e\}_{i=1}^M$. The feature dimension e is set to 512 in our model. Note that e and d are not equal, and we need a linear layer to project d to e denoted as image $\rightarrow$ text and e to d as text $\rightarrow$ image. In our model, we use image $\rightarrow$ text.

To learn visual and semantic information of common affordance from the queried image and texts, we adopt the cross-modal version of the InfoNCE loss [27] which is minimized to lead the encoders to maximally preserve the mutual information between the correctly matched pairs. The image-to-text and text-to-image contrastive losses are defined as:

$$
\begin{aligned}
l_k^{(v\to t)} &= -\log \frac{\exp(\langle \mathbf{v}_k, \mathbf{t}_k \rangle / \tau)}{\sum_{q=1}^{|Q_k|} \exp(\langle \mathbf{v}_k, \mathbf{t}_q \rangle / \tau)}, \\
l_k^{(t\to v)} &= -\log \frac{\exp(\langle \mathbf{t}_k, \mathbf{v}_k \rangle / \tau)}{\sum_{q=1}^{|Q_k|} \exp(\langle \mathbf{t}_k, \mathbf{v}_q \rangle / \tau)},
\end{aligned}
\tag{3}
$$

where $\langle \cdot, \cdot \rangle$ represents the cosine similarity between two vectors and τ is a learnable temperature parameter to scale the logits. Q_k is the sampling dictionary that provides negative samples for the positive samples. The cross-modal objective of common affordance learning is then computed as a weighted combination of the two losses averaged over all possible image-text pairs in each mini-batch:

$$
\mathcal{L}_{2D-text} = \frac{1}{M^2} \sum_{k=1}^{M^2} (\gamma l_k^{(v\to t)} + (1-\gamma) l_k^{(t\to v)}), \tag{4}
$$

where $\gamma = 0.5$ in our model.

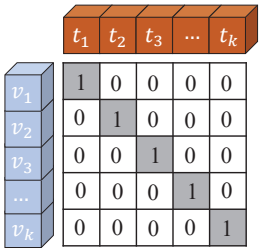
As shown in Figure 3, we considered four kinds of sampling strategies for Q_k. 1) **corr txt-img** (Figure 3a): Only the original image-text pairings in $\mathcal{A}$ are regarded as positive samples, while others are negative samples. 2) **txt-img** (Figure 3b): If the image-text pairs share a common part, we consider it as positive. 3) **img-img** (Figure 3c): By ignoring the texts, the images that have the same part categories are considered positive. 4) **txt-img + img-img** (Figure 3d): A simple combination of 2) and 3). These strategies are elaborately designed and ablated in Section 4.3.

The common affordance learning process imbues the 2D image features $\mathbf{F}^{2D}$ with three notable advantages: 1) The features are fine-grained and exhibit universal pixel-semantic alignment, owing to the encoder E_{2D} being constructed from a pretrained open-set segmentation network. 2) These features possess powerful generalization capabilities in recognizing functional interactions between parts and objects, attributed to the auxiliary image-text pairs. 3) They are well adapted for the realm of actionable part perception where the visual perceptions are dominated by object-centric images in the front view.

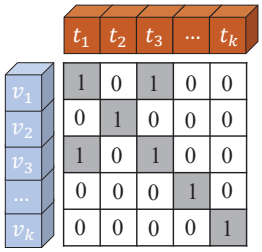
3.4. Common Affordance Distillation

The goal of our perception network is to predict 3D segmentation results for actionable parts of the objects to be manipulated. To achieve this, we propose to distill the highly generalizable knowledge about common affordance from 2D pixel features to 3D voxel features.

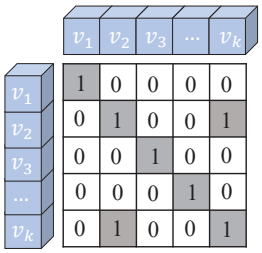
To obtain 3D voxel features, we first voxelized the point cloud P into a voxel grid and then employ a 3D Sparse UNet



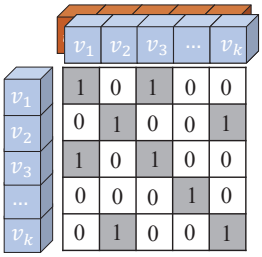
(a) **corr txt-img**. Only completely corresponding image-text pairs are considered positive samples.



(b) **txt-img**. Images and texts with the same parts are considered positive samples.



(c) **img-img**. Images with the same parts are considered positive samples.



(d) **txt-img + img-img**. Combine **txt-img** and **img-img**.

Figure 3. **Four strategies**. Experiments show that strategy **t-i** is better for improving the model's performance, and it is also the strategy adopted in our method.

to extract 3D features $F^{3D} \in \mathbb{R}^{N \times z}$, where z is our 3d feature dimension. For a 3D coordinate $\mathbf{x} \in \mathbb{R}^3$, we use $h(\cdot)$ as a mapping function to project $\mathbf{x}$ to a 2D coordinate. During training, the 3D coordinates are first multiplied with an augmented matrix M_{aug} to prevent overfitting. The values in the matrix M_{aug} are randomly generated. After mapping the 3D coordinates to 2D space, such random enhancement would cause an undesirable offset for the resulting 2D coordinates. So we also multiply the inverse matrix M_{aug}^{-1} before mapping to maintain the correspondence between 2D and 3D coordinates, as $\mathbf{x}' = h(\mathbf{x}M_{aug}^{-1})$. According to the projected coordinates we can obtain the projected 2D features of the 3D features $\mathbf{F}^{3D}$ followed by a semantic adaptor (one convolution layer) ϕ^{SA}, denoted as $\hat{\mathbf{F}}^{3D} \in \mathbb{R}^{H \times W \times d}$ Thus, the common affordance information can be distilled from 2D to 3D features via a distillation loss:

$$L_{2D-3D} = \sum_{i=1}^{H} \sum_{j=1}^{W} \mathcal{L}(\mathbf{F}_{i,j}^{2D}, \hat{\mathbf{F}}_{i,j}^{3D}), \tag{5}$$

where $\mathcal{L}$ denotes L2 loss function.

3.5. Actionable Part Perception

Through common affordance distillation, the 3D voxel features are enriched with the semantic information of actionable parts, making it discriminable in the visual-semantic cross-modal feature space. Therefore, our model can predict 3D part segmentation results by computing the cosine similarities between the 3D point-wise features and the text embeddings of all the part labels $\mathcal{C}^p$. The loss for 3D segmentation is:

$$L_{3D-text} = \mathcal{L}_{bce}(\hat{F}^{3D} E^{text}(\mathcal{C}^p)^T, Y^{3D}), \tag{6}$$

where $\mathcal{L}_{bce}$ is the binary cross-entropy loss, $Y^{3D} \in \mathbb{R}^N$ denotes the ground-truth part label for each 3D point. Then we use grouping and scoring and obtain M instance masks through non-maximum suppression (NMS), which serve as our final instance segmentation results.

3.6. Part Pose Estimation

For each predicted part segmentation mask M_i, we query its mask feature $F_{M_i}^{3D}$ from the feature F^{3D}. Then we utilize a part-oriented pose prediction network PE to regress point-wise normalized part coordinate space (NPCS) coordinates [8]. The design of NPCS loss L_{NPCS} is similar to [30]. At the inference time, we estimate the 7-dimensional rigid transformation and obtain the pose of the predicted part. As for object manipulation, the robotic arm takes the segmentation mask and pose of the target part, and then executes corresponding controlling programs to complete an operation, such as crawl, push/pull, rotate, and so on.

Finally, the overall objective for the actionable part perception framework can be formulated as:

$$\begin{aligned} L_{aux} &= \lambda_1 L_{2D-text} + \lambda_2 L_{2D-3D} + \lambda_3 L_{3D-text}, \\ L_{total} &= L_{aux} + L_{seg} + L_{NPCS}, \end{aligned} \tag{7}$$

where $\lambda_1, \lambda_2, \lambda_3$ are weights for the auxiliary losses, and L_{seg} is the part segmentation loss as in [11] without common affordance incorporation.

4. Experiments

4.1. Implementation Details and Settings

Data preparation. We use a proportion of the GAPart dataset proposed in GAPartNet [8] which is from PartNet-Mobility [35], and the rest data (from AKB-48 [16]) are not used due to the lack of 3D objects shapes. The dataset contains 8389 GAPart instances on 1045 objects with point-level semantic labels and pose annotations. There are 9 actionable parts and 23 object categories in the dataset, and we split 23 object categories into 13 seen and 10 unseen categories. Seen categories are further split into seen and unseen instances. Unseen object instances from seen and unseen categories are denoted as intra-test and outer-test, respectively. This two-level splitting enables all the part classes to exist in both seen and unseen object categories, and make a more comprehensive and fair comparison of the generalization capabilities of our method. In our experiment,

Model	Train Data	intra-test			outer-test		
		mAP	AP50	mIoU	mAP	AP50	mIoU
PointGroup [11]	PNM, AKB-48	57.3	66.8	-	21.9	26.3	-
SoftGroup [28]	PNM, AKB-48	58.5	68.8	-	22.0	27.7	-
AutoGPart [17]	PNM, AKB-48	57.2	66.3	-	25.7	30.5	-
GAPartNet [8]	PNM	60.1	71.9	75.0	25.8	33.3	31.2
Ours	PNM	**61.9**	**74.0**	**75.3**	**30.6**	**38.7**	**35.1**

Table 1. **Resutls of DisCo comparing with other methods**. PNM represents PartNet-Mobility [35]. Our model has seen a significant improvement on the outer-test, from 25.8% mAP to 30.6% mAP.

DisCo models are trained on seen object instances (also seen categories) and evaluated on the above two test datasets.

Evaluation metrics. In line with existing 3D semantic instance segmentation benchmarks in ScanNet [4] and S3DIS [1], we have opted to employ the widely accepted average precision metric to assess the performance of part segmentation. More specifically, we utilize AP50, which represents the average precision at an intersection over union threshold of 50%, to evaluate the performance of each part class as well as the overall performance. Additionally, we incorporate mAP, which calculates the average precision across IoU thresholds ranging from 50% to 95% in increments of 5%, to provide a comprehensive evaluation of the model performance. We also report the 3D mean Intersection over Union (mIoU) to evaluate part segmentation performance.

Implementation Details. We use the SAPIEN 2.0 [35] environment to render a large-scale dataset from GAPartNet objects, consisting of 2D RGB images, depth images, part semantic segmentation masks, part instance segmentation masks, and part pose annotations. Using camera intrinsic matrix, 2D RGB images, and depth images, we do back-projection to obtain dense, partial point clouds. We sample 20,000 points for each dense point cloud using Farthest-Point-Sampling (FPS). We also generate the corresponding ground truth for semantic segmentation, instance segmentation, and part pose. In our best practice, the ViT-L model of Lseg [14] is used as the backbone of the 2D image encoder E^{2D}, two Sparse UNets [9] as used as 3D backbone E^{3D} and part pose estimation network P, the CLIP [23] text encoder is used to encode texts in DisCo. Empirically, the dimension d for 2D-3D distillation is set to 32, and the loss coefficients λ_1, λ_2, λ_3 are set to 0.1, 10, and 1, respectively. During training, we use Adam optimizer [13] with a learning rate of 1e-3 and a batch size of 64.

4.2. Actionable Part Perception Results

Numerical results. Table 1 shows the quantitative comparisons between our method and existing state-of-the-art methods of 3D semantic instance segmentation, e.g., PointGroup [11] and SoftGroup [28]. We set up a baseline modified from AutoGPart [17], whose task is different from ours thus we directly combine their methods with the original PointGroup [11] pipeline for comparison. We also compare our method with GAPartNet [8], which is the first to propose 3D actionable object part segmentation. From the table, we can see that our proposed DisCo model outperforms existing methods by large margins on both intra-test and outer-test over all the evaluation metrics. Notably, the performance improvements (4.8% mAP, 5.4% AP50, and 3.9% mIoU) on the outer-test set are much more significant than those (1.8% mAP, 2.1% AP50, and 0.3% mIoU) on the intra-test set, which demonstrates that introducing common affordance could achieve remarkable generalization ability for actionable part perception models.

Qualitative results. In Figure 4, we show some visualization examples on the outer-test set. The first two rows of input point clouds data come from the outer-test in PartNet-Mobility [35], and the last two rows are from AKB-48 [16]. “pc", “pred", “sem", “gt", “ins", “bbox" stand for point cloud, prediction, semantic segmentation, ground truth, instance segmentation, prediction bounding box. Our model predicts accurate results of actionable part perception even on the tiny parts like buttons on the remote control. When directly tested on new objects from the AKB-48 dataset where the RGB style is different from training data, our model still delivers impressive performance.

4.3. Ablation Studies

We ablate the key factors of our method in Table 2. Numerical results are reported on mAP(%), which represents the performance of instance segmentation, and the higher the better. “intra” and “outer” represent the performance on seen categories unseen object instances and unseen categories unseen object instances, respectively. In general, we pay more attention to the performance on “outer” since it measures the generalization ability of models. In the following ablation studies, we employ the best practices of the model settings except for the ablated factor.

Visual and semantic information of common affordance. DisCo introduces both visual and semantic information about common affordance into the 3D point features. We conduct ablation studies to evaluate the effectiveness of both types of information in Table 2a. We first build a vanilla model (denoted as “bare”) by disabling the common affordance

2D-3D	intra	outer
bare	54.47	21.95
+ visual	54.58	24.62
+ semantic	**60.51**	**27.60**

(a) **Visual and semantic information of common affordance** help improve the generalization ability of our model.

type	intra	outer
Cos	57.45	**27.79**
MSE	**60.50**	27.60

(b) **Different implementation of** $\mathcal{L}_{2D-3D}$. Mean square error loss is more effective than cosine similarity loss.

dim	intra	outer	iter/s
16	57.92	26.92	0.54
32	**60.50**	27.60	0.52
64	59.15	**27.93**	0.48

(c) **Feature dimension**. We take 32 as the feature dimension based on a trade-off between performance and efficiency.

ct strategy	intra	outer
-	**60.51**	27.60
corr i-t	58.22	26.60
i-t	60.05	**28.53**
i-i	56.87	25.65
i-t + i-i	58.41	26.40

(d) **Contrastive loss strategy**. Strategy img-txt is more suitable for our model.

case	intra	outer
image → text	**60.05**	**28.53**
text → image	57.71	28.49

(e) **Mapping direction**. Mapping image features into text feature space is more efficient.

scale	intra	outer
0.5x	57.97	28.49
1.0x	60.05	28.53
2.0x	**61.90**	**30.64**
3.0x	61.30	28.94

(f) **Scale of auxiliary data**. A proper scale of the generated data is more effective.

Table 2. **Ablation studies on key factors of DisCo**. We report mAP(%) on both the "intra-test"(seen object category and unseen object instance) and "outer-test"(unseen object category and unseen object instance). Best practice settings are marked in gray .

discovering. The 3D features generated by E^{3D} are directly fed into the segmentation head to predict part segmentation results, which is supervised by the ground-truth 3D point labels via BCE loss. Then we introduce the visual common affordance information and add the L_{2D-3D} loss into model training, resulting in a significant improvement on the "outer" test set, from 21.95% mAP to 24.62% mAP. Furthermore, we validate the benefit of the semantic information of common affordance by enabling the $L_{2D-text}$ and $L_{3D-text}$ losses, obtaining consistent improvements on both the "intra-val" (from 54.58% to 60.51% mAP) and "outer-val" (from 24.62% to 27.60% mAP).

Different implementation of L_{2D-3D}. We compare two common implementations of the 2D-3D feature distillation loss L_{2D-3D}, they are, the cosine similarity loss and the MSE loss. From Table 2b we can see that the MSE loss achieves a better performance on "intra" (60.50% mAP v.s. 57.45% mAP), but a small drop on "outer". This is because the MSE loss is more suitable for regression tasks, which is consistent with the goal of L_{2D-3D}.

Different values of feature dimension d. During common affordance distillation, we use a domain adaptor and a semantic adaptor to project the original 2D image features and 3D features to have equal dimension d. Larger dimensions would lead to better performance but also an undesirable increment in the computing resource overhead. To find a balance between performances and computation costs, we tried different dimension values, such as 16, 32, and 64, see Table 2c. We can see that from 32 to 64, the performance on outer-test improved by 0.33% mAP, but the iteration speed slowed down by 8.3%. When the total number of training iterations increases, this will cause considerable consumption. To balance performance and efficiency, we set d to 32 in our model.

Contrastive loss strategy. We conducted ablation experiments on the four contrastive loss strategies mentioned in Section 3.3. Not using contrastive loss $L_{2D-text}$ is denoted as "-". "i" and "t" represent "img" and "txt" respectively. In Table 2d, we show that the contrast loss strategy that sets the image text as a positive sample as long as there are the same parts has the best effect on the model, increasing 27.6% mAP to 28.53% mAP on outer-test. Although the mAP score on the intra-test has declined, this is an acceptable slight decay (0.46% mAP), because we are more concerned with the outer-test. Note that using image-to-image matching can result in performance degradation (from 28.53% to 25.63%). Parts of an object occupy a small space in an image, therefore the image features usually have a large amount of redundant information, making it difficult to distinguish them.

Mapping direction. Table 2e shows that using image → text works better than text → image. The former is 0.04% mAP higher than the latter on outer-test while 2.34% on intra-test. The reason may be that we prefer to align each text feature token to the image feature token rather than align each image feature token to the text feature token. In this way, our model can learn open-world ability from the pretrained text encoder.

Scale of generated data. We generate image-text pairs of different scales to see the effect of the data scale on the performance. Based on the number of point clouds, generate auxiliary data with different ratios, such as "0.5x", "1.0", "2.0x", and "3.0x". From Table 2f we can see that increasing the scale of the generated images from "0.5x" to "2.0x" can improve the generalization ability of the model. Nevertheless, when more images are generated and the ratio reaches "3.0x" the performances decrease on both the intra and the outer test set. We believe that unbalanced proportions introduce more noise and make the model more inclined to learn 2D's ability. More results of generated images can be found in the appendix.



Figure 4. **Qualitative results**. The first two rows of input point clouds data come from the outer-test in PartNet, and the last two rows are from AKB-48. "pc", "pred", "sem", "gt", "ins", "bbox" stand for point cloud, prediction, semantic segmentation, ground truth, instance segmentation, prediction bounding box. Best viewed in color.

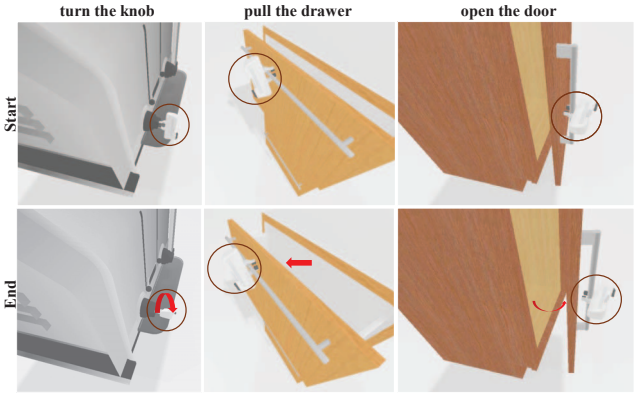


Figure 5. **Manipulation in Simulator**. Input the prediction results of the pose prediction network, and the robot arm can locate the position of the part and perform the corresponding operation according to the part category.

4.4. Part based Manipulation.

After obtaining accurate part perception results, we calculate the poses of objects' parts via the Pose Estimation network. Then we move the robot arm toward the target part, turn the gripper in the direction suitable for grabbing, and then close the gripper. Finally, we move the gripper along the GAPart part classes' trajectories toward the target position. The results in Figure 5 show successful controlling operations of object manipulation including "turn the knob", "pull the drawer", and "open the door".

5. Conclusion

We introduce DisCo, a method that queries, learns, and distills common affordance to create robust and distinctive representations for perceiving actionable parts. Leveraging a Large Language Model and a diffusion model, DisCo harnesses rich visual and functional interaction details to capture common affordance information among objects' actionable parts. This information is encoded into fine-grained image features and then distilled from 2D to 3D point features. Experimental results showcase DisCo's exceptional generalization capabilities and highlight its substantial impact on enhancing actionable part perception, thereby improving open-world object manipulation.

Acknowledgements This work was supported in part by National Science and Technology Major Project (2020AAA0109704), National Science Foundation of China Grant No. 62476293, National Key Research and Development Program (Grant 2022YFE0112500), National Natural Science Foundation of China (Grants 62101607), Guangdong Outstanding Youth Fund (Grant No. 2021B1515020061), Shenzhen Science and Technology Program (Grant No. GJHZ20220913142600001), Nansha Key RD Program under Grant No.2022ZD014, The Major Key Project of PCL (No. PCL2024A04).

References

[1] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1534–1543, 2016. 6

[2] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2

[3] Wei Chen, Xi Jia, Hyung Jin Chang, Jinming Duan, Linlin Shen, and Ales Leonardis. Fs-net: Fast shape-based network for category-level 6d object pose estimation with decoupled rotation mechanism. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1581–1590, 2021. 3

[4] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 6

[5] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. Pla: Language-driven open-vocabulary 3d scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7010–7019, 2023. 2

[6] Samir Yitzhak Gadre, Kiana Ehsani, and Shuran Song. Act the part: Learning interaction strategies for articulated object part discovery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15752–15761, 2021. 3

[7] Haoran Geng, Ziming Li, Yiran Geng, Jiayi Chen, Hao Dong, and He Wang. Partmanip: Learning cross-category generalizable part manipulation policy from point cloud observations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2978–2988, 2023. 1

[8] Haoran Geng, Helin Xu, Chengyang Zhao, Chao Xu, Li Yi, Siyuan Huang, and He Wang. Gapartnet: Cross-category domain-generalizable object perception and manipulation via generalizable and actionable parts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7081–7091, 2023. 1, 2, 3, 5, 6

[9] Benjamin Graham and Laurens Van der Maaten. Submanifold sparse convolutional networks. *arXiv preprint arXiv:1706.01307*, 2017. 3, 6

[10] Zeyu Hu, Xuyang Bai, Jiaxiang Shang, Runze Zhang, Jiayu Dong, Xin Wang, Guangyuan Sun, Hongbo Fu, and Chiew-Lan Tai. Vmnet: Voxel-mesh network for geodesic-aware 3d semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15488–15498, 2021. 2

[11] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Pointgroup: Dual-set point grouping for 3d instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and Pattern recognition*, pages 4867–4876, 2020. 5, 6

[12] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language embedded radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19729–19739, 2023. 1

[13] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[14] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven semantic segmentation. In *International Conference on Learning Representations*, 2022. 3, 6

[15] Kunhao Liu, Fangneng Zhan, Jiahui Zhang, Muyu Xu, Yingchen Yu, Abdulmotaleb El Saddik, Christian Theobalt, Eric Xing, and Shijian Lu. 3d open-vocabulary segmentation with foundation models. *arXiv preprint arXiv:2305.14093*, 2023. 2

[16] Liu Liu, Wenqiang Xu, Haoyuan Fu, Sucheng Qian, Qiaojun Yu, Yang Han, and Cewu Lu. Akb-48: A real-world articulated object knowledge base. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14809–14818, 2022. 2, 5, 6

[17] Xueyi Liu, Xiaomeng Xu, Anyi Rao, Chuang Gan, and Li Yi. Autogpart: Intermediate supervision search for generalizable 3d part segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11624–11634, 2022. 2, 6

[18] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 909–918, 2019. 2

[19] Chuanruo Ning, Ruihai Wu, Haoran Lu, Kaichun Mo, and Hao Dong. Where2explore: Few-shot affordance learning for unseen novel categories of articulated objects. *arXiv preprint arXiv:2309.07473*, 2023. 1

[20] Songyou Peng, Kyle Genova, Chiyu Jiang, Andrea Tagliasacchi, Marc Pollefeys, Thomas Funkhouser, et al. Openscene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 815–824, 2023. 2

[21] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 2

[22] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 2

[23] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 3, 6

[24] Mohammad Mahfujur Rahman, Clinton Fookes, Mahsa Baktashmotlagh, and Sridha Sridharan. Multi-component image translation for deep domain generalization. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 579–588. IEEE, 2019. 2

[25] Adam Rashid, Satvik Sharma, Chung Min Kim, Justin Kerr, Lawrence Yunliang Chen, Angjoo Kanazawa, and Ken Goldberg. Language embedded radiance fields for zero-shot task-oriented grasping. In *7th Annual Conference on Robot Learning*, 2023. 1

[26] Ayça Takmaz, Elisabetta Fedele, Robert W Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. Openmask3d: Open-vocabulary 3d instance segmentation. *arXiv preprint arXiv:2306.13631*, 2023. 2

[27] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 4

[28] Thang Vu, Kookhoi Kim, Tung M Luu, Thanh Nguyen, and Chang D Yoo. Softgroup for 3d instance segmentation on point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2708–2717, 2022. 6

[29] Chen Wang, Roberto Martín-Martín, Danfei Xu, Jun Lv, Cewu Lu, Li Fei-Fei, Silvio Savarese, and Yuke Zhu. 6-pack: Category-level 6d pose tracker with anchor-based keypoints. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10059–10066. IEEE, 2020. 3

[30] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2642–2651, 2019. 3, 5

[31] Jiaze Wang, Kai Chen, and Qi Dou. Category-level 6d object pose estimation via cascaded relation and recurrent reconstruction networks. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4807–4814. IEEE, 2021. 3

[32] Qianxu Wang, Haotong Zhang, Congyue Deng, Yang You, Hao Dong, Yixin Zhu, and Leonidas Guibas. Sparsedff: Sparse-view feature distillation for one-shot dexterous manipulation. *arXiv preprint arXiv:2310.16838*, 2023. 1

[33] Yixuan Wang, Zhuoran Li, Mingtong Zhang, Katherine Driggs-Campbell, Jiajun Wu, Li Fei-Fei, and Yunzhu Li. D^3 fields: Dynamic 3d descriptor fields for zero-shot generalizable robotic manipulation. *arXiv preprint arXiv:2309.16118*, 2023. 1

[34] Yijia Weng, He Wang, Qiang Zhou, Yuzhe Qin, Yueqi Duan, Qingnan Fan, Baoquan Chen, Hao Su, and Leonidas J Guibas. Captra: Category-level pose tracking for rigid and articulated objects from point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13209–13218, 2021. 3

[35] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, et al. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11097–11107, 2020. 2, 5, 6

[36] Hengshuang Zhao, Li Jiang, Chi-Wing Fu, and Jiaya Jia. Pointweb: Enhancing local neighborhood features for point cloud processing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5565–5573, 2019. 2

[37] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pages 561–578. Springer, 2020. 2

[38] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4490–4499, 2018. 2