Supplementary for "3D AffordanceNet: A Benchmark for Visual Object Affordance Understanding"

This supplementary material provides additional dataset visualization, qualitative results, technical details, full question list and the interface of annotation tool.

In Sec. A we present all questions visible to annotators during data annotation procedure. Data annotation agreement is discuss in Sec.B. We further provide more ground-truth visualizations for each shape category in 3D AffordanceNet dataset in Sec. C. The Sec. D presents more qualitative examples of full-shape, partial-view and rotation-invariant affordance estimation results with PointNet++ and DGCNN as backbones. In Sec. E we describe more details about neural network architecture and training parameters. We visualize the annotation interface and the main components of our web-based annotation tool in Sec. F. Then we explore the performance gain benefit from fine-tuning in Sec. G. At last we demonstrate why a truly functional understanding of object affordance requires learning and prediction in the 3D physical domain in Sec. H.

A. Complete List of Questions

We list all questions that the annotation interface displays to annotators. The complete question list is shown in Tab. 1. The questions that the annotation interface proposes directly determine how the annotators understand the affordance, thus we carefully define the questions for each affordance.

B. Data Annotation

We demonstrate that the human perceived affordance often do not fully overlap with the individual part specified in PartNet dataset. To investigate the relation between parts and affordances, we report the maximal IoU between each affordance and a best combination of parts (most fine-grained level) in Tab.2. The low IoU indicates that it is impossible to combine several parts to derive affordance, suggesting the necessity to label affordance from scratch.

We have 3 annotators to label each object, which gives certain diversities and partially addresses the issue of ambiguities. We report the inconsistency as average variance across 3 annotators in Tab.2. The relatively smaller variance of each affordance indicates that there exists patterns in the annotations, and models can therefore learn affordance from data.

C. More Ground-Truth Visualizations

We present more ground-truth visualization in Fig. 6 and Fig. 7. From the visualization of ground-truth, we can observe that the human perceived affordances often do not fully overlap with the individual parts specified in PartNet dataset, therefore it justifies the need to annotate affordance separately from existing part annotations.

D. More Qualitative Examples

We present more qualitative examples for full-shape, partial-view and rotation-invariant affordance estimation experiments with both PointNet++ and DGCNN as backbone in Fig. 1, 2, 3 and 4.

The estimation results from PointNet++ and DGCNN are quite interesting. The predicted affordance locations from the two networks are close while the confidences of the points belonging to specific affordances have different tendencies. In many cases, *e.g. grasp* for *bag* and *press* for *laptop*, PointNet++ tends to predict scores with low confidence which will cause more false-negative predictions while DGCNN predicts scores more aggressively, leading to more false-positive examples.

E. Training Details

In this section we describe more details of training procedure. We conduct all experiments using the segmentation branch of PointNet++ and DGCNN as shared backbones.

In specific, the dimension of point-wise features by PointNet++ and DGCNN are 128 and 256, respectively. We formulate affordance estimation as a binary classification problem, therefore, we set up classification heads for each affordance category, in total there are 18 classification heads which have the same architecture with different parameters. Specifically, the classification head for each affordance category consists of $FC(m, 128), FC(128, 1)$, where the function $FC(x, y)$ denotes a fully connected layer that takes x dimension vectors as inputs and outputs y dimension vectors, the number m denotes the dimension

Object	Affordance	Question
Bed	Lay	If you were to lie on this bed, which points would you lie on the bed?
	Sit	If you were to sit on this bed, at which points on the bed would you sit?
	Support	If you put something on this bed, at which points on the bed would you put it?
Bag	Grasp	If you want to grab the bag, at which points will your palm position be?
	Lift	If you want to lift the bag, at which point is your finger most likely to carry the bag?
	Contain	If you package things into the bag, at which points in this bag would you put?
	Open	If you want to open the bag, from which points of the package would you open it?
Bottle	Grasp	If you grab the bottle, at which points on the bottle handle will your palm touch?
	Wrap-Grasp	If you wrap-grasp the bottle, at which points on the bottle wall will your palm touch?
	Contain	If you pour water into the bottle, which points will the water first touch when it falls into the bottle?
	Open	If you want to open this bottle, from which points on the cap would you open it?
	Pour	Suppose there is water in the bottle, and you want to pour the water out of the bottle. From which points on the bottle will the water flow out?
Bowl	Wrap-Grasp	If you wrap-grasp the bowl, at which points on the bowl wall will your palm touch?
	Contain	If you want to put something in the bowl, at which points in the bowl would you put it?
	Pour	Suppose there is water in the bowl, and you want to pour the water out of the bowl. From which points on the bowl will the water flow out?
Chair	Sit	If you were sitting on this chair, on which points would you sit?
	Support	If you want to put something on the chair, at which points on the chair would you put it?
	Move	If you want to move this chair, at which points on the chair will you exert force?
Clock	Display	If you want to look at the time, which points on this clock would you look at?
Dishwasher	Contain	If you want to load things in the dishwasher, at which points in the dishwasher would you put the things?
	Open	If you want to open this dishwasher, from which points on the dishwasher door would you open it?
Display	Display	If you look on the screen, which points on the screen will you look at?
Door	Push	If you want to push the door, at which points on the door will your palm touch?
	Open	If you were to open the door, from which points on the door would you open it?
	Pull	If you want to pull the door, which points on the door will you pull with your finger?
Earphone	Grasp	If you want to grab this earphone, where will your palm position be?
	Listen	If you want to listen to music with headphones, which points on the headphones will point to your ears?
Faucet	Grasp	If you want to grab this faucet, which points on the faucet will your palm touch?
	Open	If you want to boil water, at which points on the tap would you open the water valve?
Hat	Grasp	If you want to grab this hat, which points on the hat will your palm touch?
	Wear	If you want to wear this hat, which points on the hat will make contact with your head?
Keyboard	Press	If you want to press keys on the keyboard, which points on the keyboard would you press?
Knife	Grasp	If you want to grab this knife, at which points on the handle will your palm touch?
	Cut	If you want to cut something with this knife, which points on the blade will come into contact with the object?
	Stab	If you use this knife to poke an object, which points on the blade will come into contact with the object?
Laptop	Display	If you look on the computer screen, which points on the screen will you look at?
	Press	If you want to press keys on a computer keyboard, which points on the keyboard would you press?
Microwave	Open	If you want to open the microwave, from which points on the microwave door would you open it?
	Contain	If you put something in the microwave, at which points in the microwave would you put the object?
Mug	Pour	Suppose there is water in the mug, and you want to pour the water out of the mug. From which points on the mug will the water flow?
	Contain	If you pour water into the mug, which points will the water first touch when it falls into the mug?
	Wrap-Grasp	If you wrap-grasp this mug with your hand, which points on the mug will your palm touch?
	Grasp	If you grab this mug, which points on the mug handle will your palm touch?
Refrigerator	Contain	If you put things in the refrigerator, at which points in the refrigerator would you put?
	Open	If you want to open the refrigerator, from which points on the refrigerator door would you open it?
Scissors	Grasp	If you want to grab this scissors, which points on the handle of the scissors will your palm touch?
	Cut	If you want to use scissors to cut something, which points on the scissors blade will contact the object?
	Stab	If you poke an object with this pair of scissors, which points on the blade will come into contact with the object?
StorageFurniture	Contain	If you want to put something in the cabinet, at which points in the cabinet would you put it?
	Open	If you want to open this cabinet, from which points on the cabinet door would you open it?
Table	Support	If you want to put something on the table, at which points on the table would you put the object?
	Move	If you want to move this table, at which points on this table will you exert your strength
TrashCan	Contain	If you put trash in the trash can, which points will the trash drop first touch?
	Pour	If you want to dump out the trash in the trash can, at which points on the trash can will the trash slip out?
	Open	If you want to open the lid of this trash can, from which points on the trash can you open it?
Vase	Contain	If you pour water into the vase, which points will the water first touch when it falls into the vase?
	Pour	Suppose there is water in the vase, and you want to pour the water out of the vase. From which points on the vase will the water flow out?
	Wrap-Grasp	If you wrap-grasp this vase with your hands, which points on the vase will your palm touch?

Table 1. The complete list of questions that the annotation interface proposes to annotators

Affordance	Grasp	Lift	Contain	Open	Lay	Sit	Support	Wrap.	Pour
IOU	22.7	35.9	13.8	28.3	26.4	18.8	14.4	6.8	14.2
Variance	0.007	0.004	0.002	0.003	0.005	0.004	0.005	0.005	0.004
Affordance	**Display**	**Push**	**Pull**	**Listen**	**Wear**	**Press**	**Move**	**Cut**	**Stab**
IOU	39.8	13.3	34.1	17.4	7.3	29.7	12.3	16	3.5
Variance	0.005	0.005	0.002	0.002	0.006	0.18	0.009	0.004	0.001

Table 2. IOU between each affordance and a best combination of parts. Variance of each affordance category. Numbers of IOU are in %.

of point-wise features by the shared backbones (in our case, it will be 128 for PointNet++ and 256 for DGCNN). In practice, the first FC is followed by a Batch Normalization layer.

For PointNet++, we set the training learning rate 0.001 and optimize the parameters with Adam optimizer, the learning rate is reduced by half every 20 epochs, we train the network for 200 epochs, the batch size is 16. The weight decay for Adam optimizer is set to 1e-8. For DGCNN, we set the learning rate to 0.1 and optimize the parameters with SGD optimizer, the momentum and weight decay for SGD are set as 0.9 and 1e-4, respectively. We use a cosine anneal-

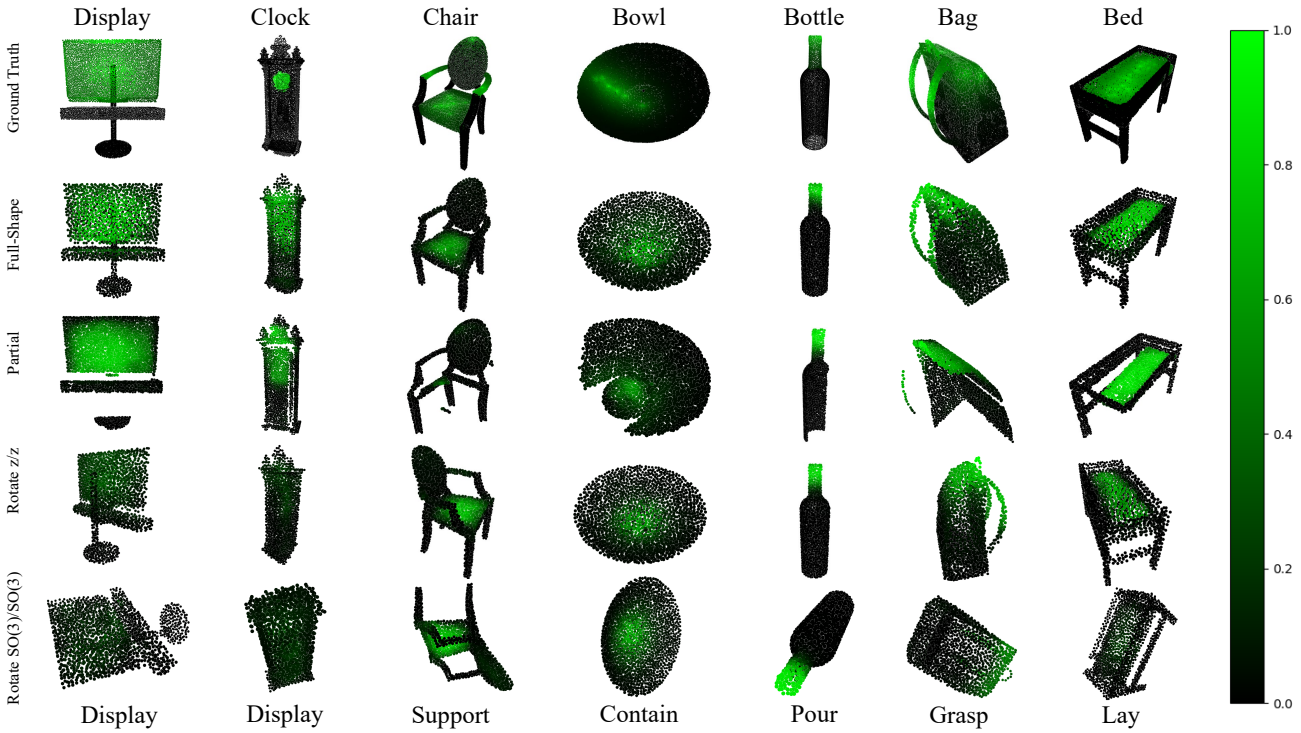

Figure 1. Qualitative results for affordance estimation from PointNet++(1/2). The top row shows the ground-truth. The second row shows the full-shape estimated results, the third row shows the partial-view estimated result, the fourth and the bottom row show the z/z and $SO(3)/SO(3)$ rotation-invariant estimated results, respectively. All results come from PointNet++. The top words indicate the semantic category of each column and the bottom words indicate the affordance category. The greener the color of the points, the higher the confidence about specific affordance types. *Wrap.* is the abbreviation of *Wrap-Grasp.*

ing algorithm to adjust the learning rate where the algorithm can be described as followed:

$$\eta_t = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min})(1 + cos(\frac{T_{cur}}{T_{max}}\pi)) \quad (1)$$

where η_t is the adjusted learning rate, η_{min} is the minimum learning rate, η_{max} is set to the initial learning rate, T_{cur} is the number of current epochs. We set the batch size to 16 and train the network for 200 epochs.

Particularly, for semi-supervised affordance estimation, we use DGCNN as shared backbone and follow the training strategies described above. We set the batch size to 16, 8 for labeled data and 8 for unlabeled data. We set the ξ and ϵ of Virtual Adversarial Training to 1e-6 and 2.0, which are the default hyper-parameters described in its paper. We calculate the virtual adversarial direction in 1 iteration which is recommended by the original paper. We implement all experiments with PyTorch.

F. GUI Interface for Annotation

We show the GUI interface of web-based annotation tool in Fig. 5. We manually modify the annotation system released by PartNet fit our requirements. We color the parts of shapes according to the pre-defined colormap in PartNet dataset.

The annotators can observe the geometric information of shapes and are able to freely translate, rotate and change the scale of shapes in **3D GUI**. In **Question Workflow**, the annotation interface asks the annotators some questions to guide them to select keypoints on the surface of shapes. From **Supported Affordance**, the annotators can check the supported affordances that are determined by selecting the corresponding affordances in **Affordance List**.

G. PointContrast Fine-Tune

	mAP	mAUC	aIOU	MSE		mAP	mAUC	aIOU	MSE
Fine-Tune	**47.4**	**86.3**	**19.7**	**0.063**	Scratch	45.9	85.8	19.1	0.064

Table 3. The full-shape affordance estimation performance comparison between the U-Net fine-tuned on our dataset and the U-Net trained from scratch.

In 2D vision, in order to boost the performance, it is popular to fine-tune a network on the smaller target set where the network was pre-trained on a rich source set. Recently, PointContrast shows that by pre-training the network on the ScanNet dataset using contrastive learning, the pre-trained

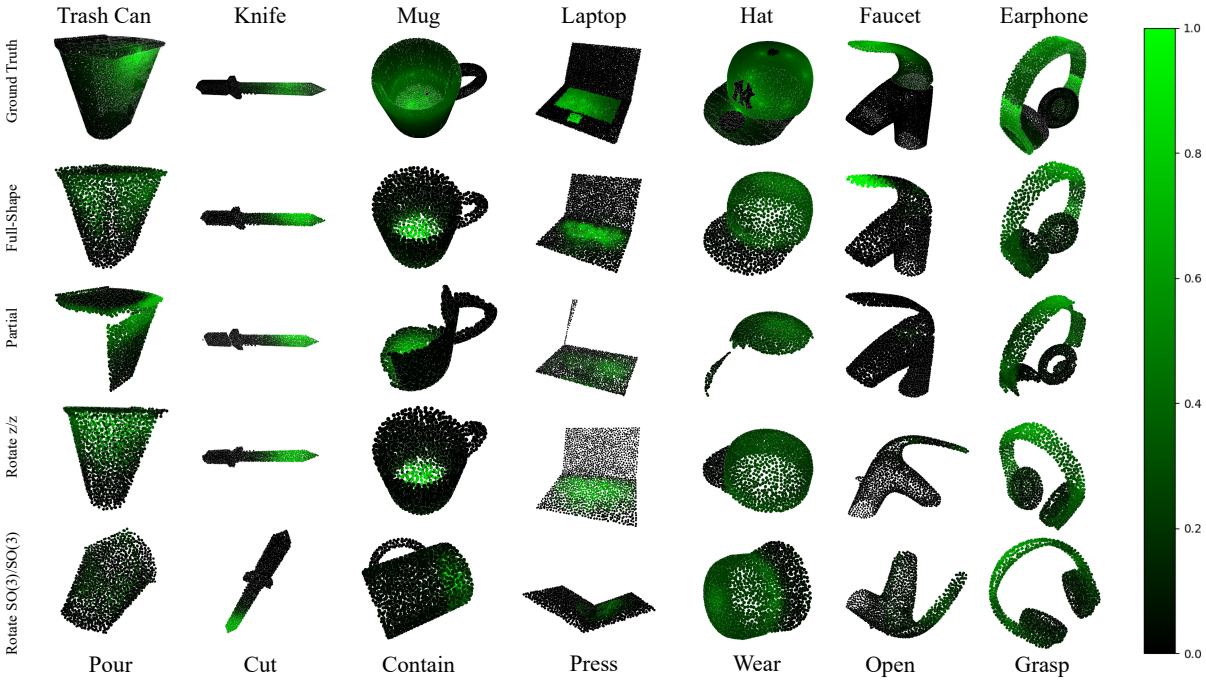

Figure 2. Qualitative results for affordance estimation from PointNet++(2/2). The top row shows the ground truth. The second row shows the full-shape estimated results, the third row shows the partial-view estimated result, the fourth and the bottom row show the z/z and $SO(3)/SO(3)$ rotation-invariant estimated results, respectively. All results come from PointNet++. The top words indicate the semantic category of each column and the bottom words indicate the affordance category. The greener the color of the points, the higher the confidence about specific affordance types. *Wrap.* is the abbreviation of *Wrap-Grasp.*

network can achieve the state-of-the art performances via fine-tuning on several downstream tasks. To explore the opportunity of boosting performances of affordance estimation by fine-tuning the pre-trained network on our 3D AffordanceNet dataset, we utilize the U-Net architecture and the pre-trained weight provided by PointContrast. We then fine-tune the network using SGD optimizer with learning rate=0.1, weight decay=1e-4 and momentum=0.9 for 60 epochs. The loss function that we use to fine-tune the network is the same as the one proposed in the Section 4.1. We also train the network straightly from scratch with the same network architecture for the comparison.

Tab. 3 reports the performances of both fine-tuned and trained-from-scratch U-Net on full-shape affordance estimation task. The results show that the performances of the fine-tuned one surpass the one that is training from scratch, meaning that the network can benefit from the pre-training on a rich source dataset during the fine-tune process on the affordance estimation task, which may also works for the other networks such as PointNet++ and DGCNN.

H. Affordance Understanding in 3D

Previous works on affordance understanding focus on learning affordances in 2D or 2.5D domain, however, many types of affordance are related to functional attributes of objects, and the relevant actions can only be accomplished given 3D affordance reasoning on the object surface. For example, a successful grasp of mug replies on inference of surface grasp points (i.e., prediction of the grasp affordance) that may be self-occluded in a single-view observation (i.e., 2.5D). Annotated 3D affordance data facilitate reasoning on the complete object surface.

	mAP	AUC	aIOU		mAP	AUC	aIOU
P 3DV	48.0	87.4	19.3	**D 3DV**	46.4	85.5	17.8
P MTV	45.1	84.4	16.6	**D MTV**	41.6	82.3	13.4
P SGV	35.0	77.8	12.9	**D SGV**	35.0	78.8	11.5

Table 4. The comparisons between 3D and 2.5D. P and D refer to PointNet++ and DGCNN respectively.

To quantify the benefit, we conduct the following experiments based on our dataset. We randomly sample one single view (2.5D) from each object for training, namely single-view partial (SGV), and randomly sample 4 views from each object, namely multi-view partial (MTV), then we test the SGV/MTV models on full-shape data. We compare SGV and MTV with training on full 3D data (3DV). Results in Tab. 4 verify our analysis. It is worth noting that the ground-truth of single view (2.5D) also relies on 3D annotation.

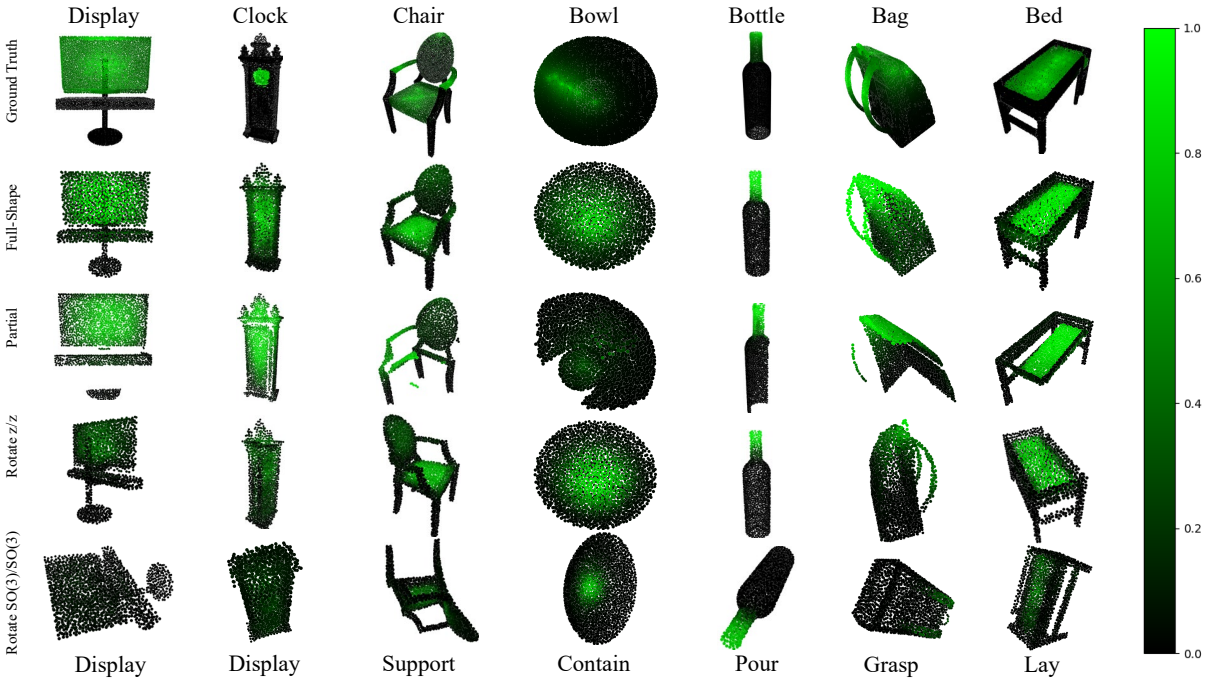

Figure 3. Qualitative results for affordance estimation from DGCNN(1/2). The top row shows the ground truth. The second row shows the full-shape estimated results, the third row shows the partial-view estimated result, the fourth and the bottom row show the z/z and $SO(3)/SO(3)$ rotation-invariant estimated results, respectively. All results come from DGCNN. The top words indicate the semantic category of each column and the bottom words indicate the affordance category. The greener the color of the points, the higher the confidence about specific affordance types. *Wrap.* is the abbreviation of *Wrap-Grasp.*

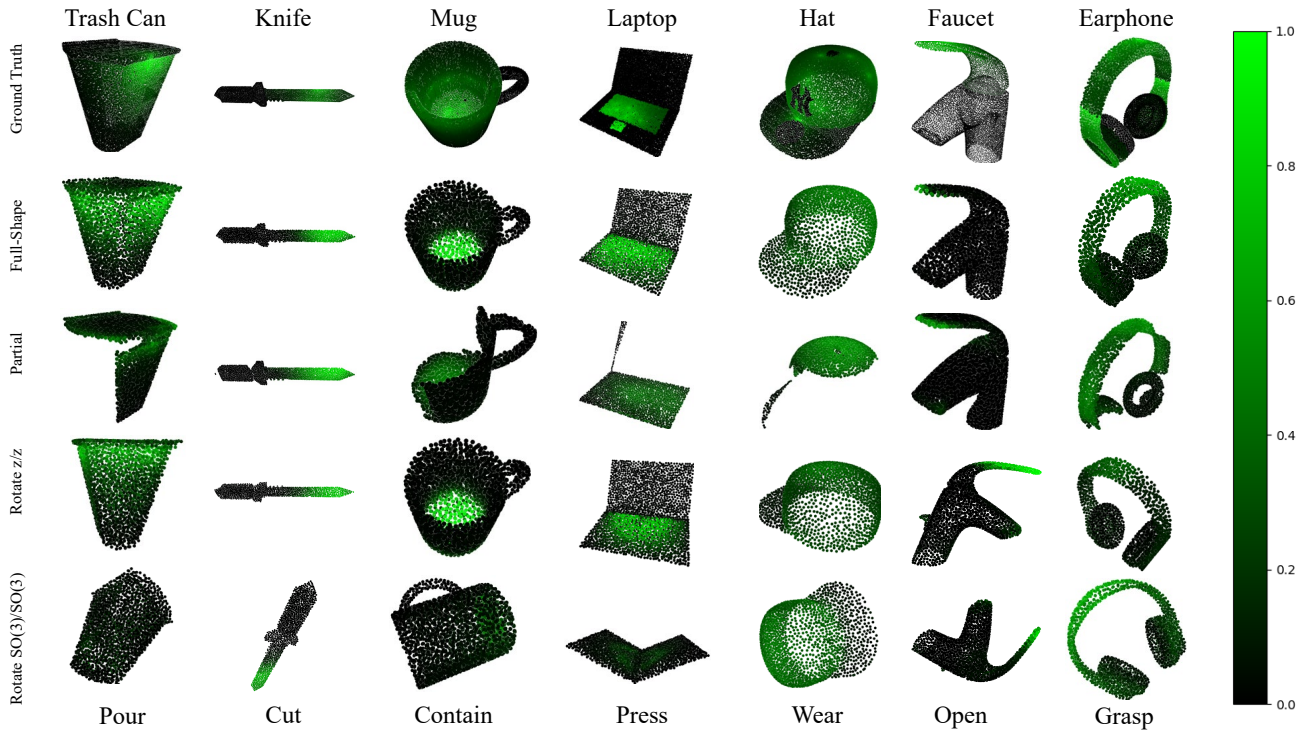

Figure 4. Qualitative results for affordance estimation from DGCNN(2/2). The top row shows the ground truth. The second row shows the full-shape estimated results, the third row shows the partial-view estimated result, the fourth and the bottom row show the z/z and $SO(3)/SO(3)$ rotation-invariant estimated results, respectively. All results come from DGCNN. The top words indicate the semantic category of each column and the bottom words indicate the affordance category. The greener the color of the points, the higher the confidence about specific affordance types. *Wrap.* is the abbreviation of *Wrap-Grasp.*

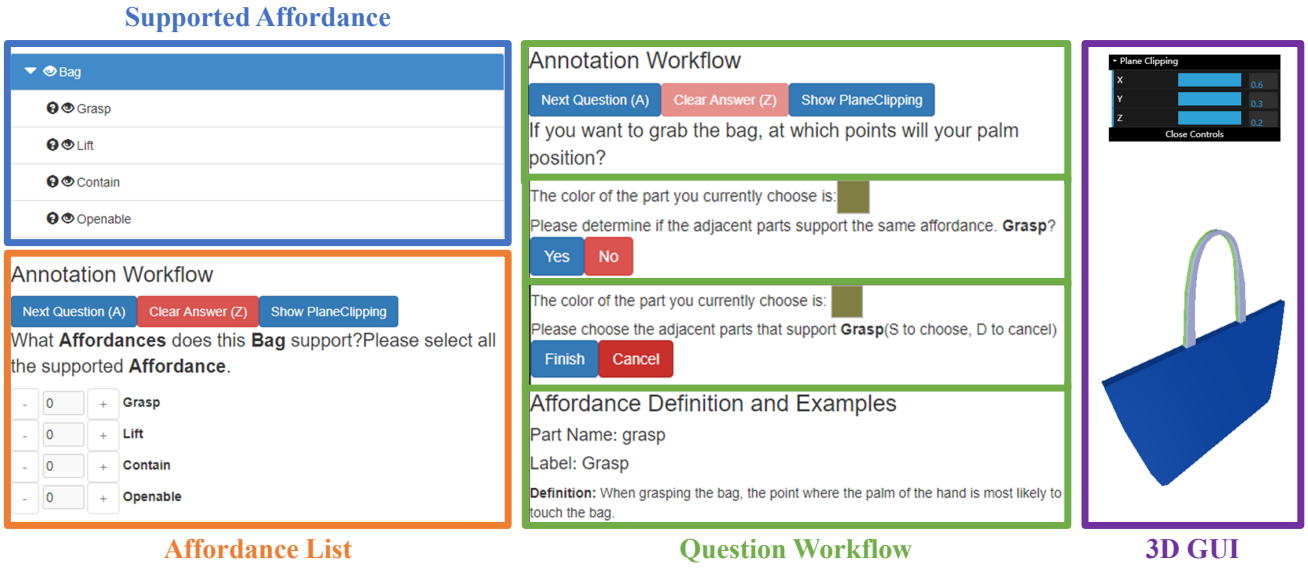

Figure 5. The annotation interface of our web-based annotation tool. We show the GUI and main component of the annotation interface.

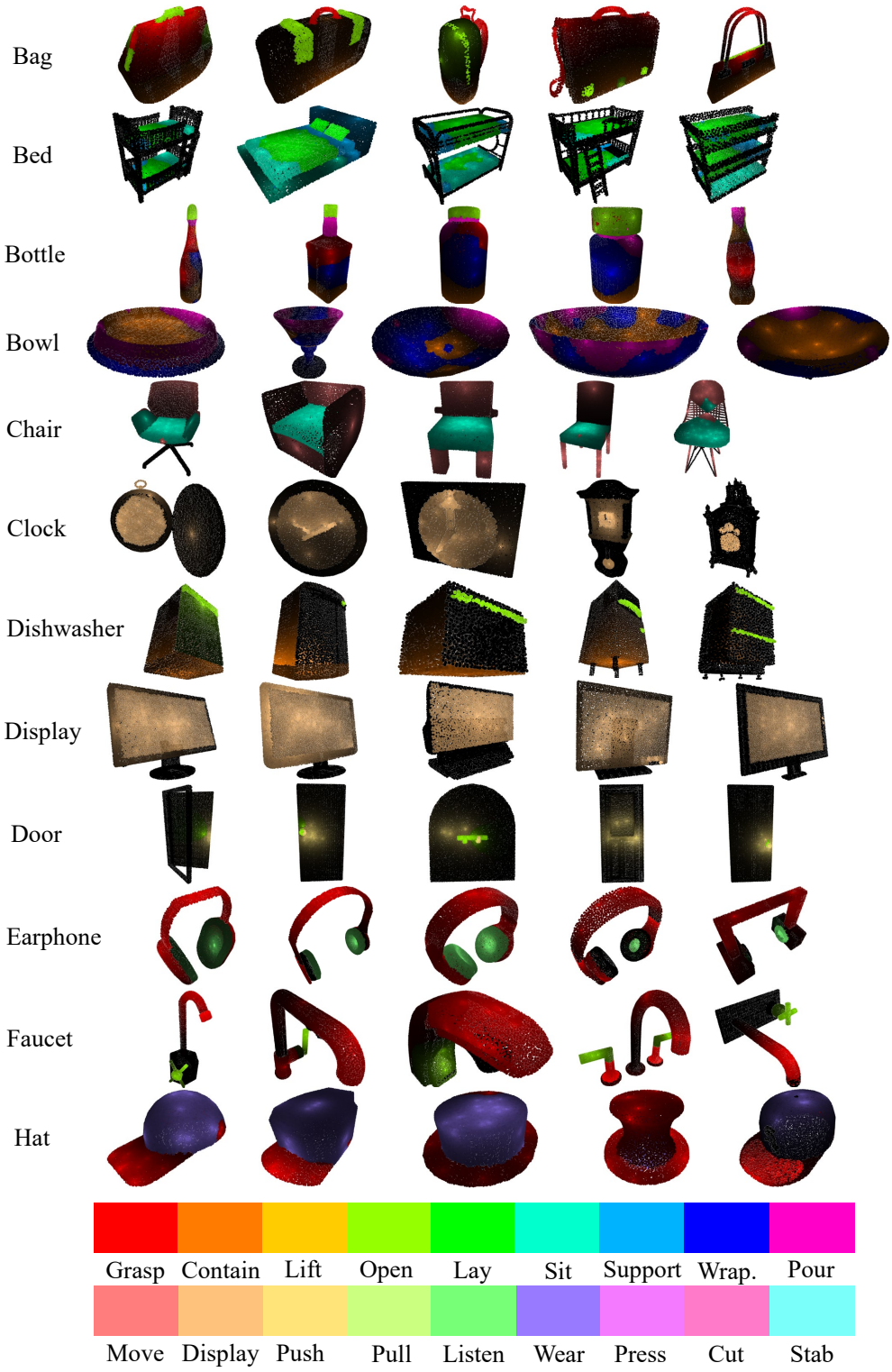

Figure 6. Ground Truth data visualization(1/2).

Figure 7. Ground Truth data visualization(2/2).