Omni6D: Large-Vocabulary 3D Object Dataset for Category-Level 6D Object Pose Estimation - Supplementary Materials -

Mengchen Zhang[1,2], Tong Wu[3], Tai Wang[2], Tengfei Wang[2], Ziwei Liu[4], and Dahua Lin[2,3]

[1] Zhejiang University, Zhejiang, China
[2] Shanghai Artificial Intelligence Laboratory, Shanghai, China
[3] The Chinese University of Hong Kong, Hong Kong SAR
[4] Nanyang Technological University, Singapore
{zhangmengchen,wangtai,wangtengfei}@pjlab.org.cn,
{wt020,dhlin}@ie.cuhk.edu.hk, ziwei.liu@ntu.edu.sg

A Overviews

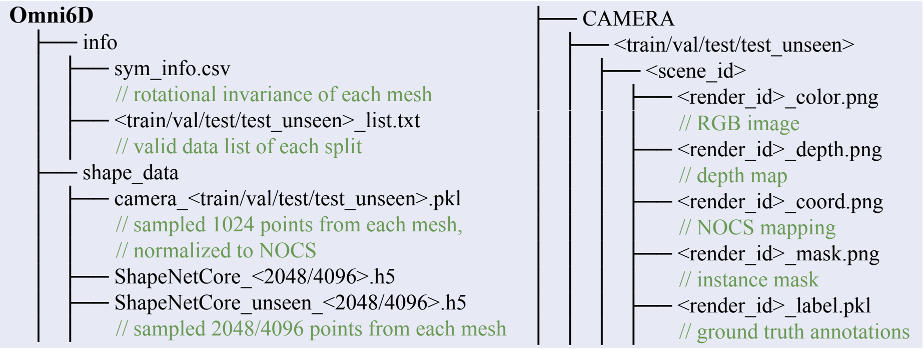
In the supplementary materials, we delve deeper into our research, offering a comprehensive exploration of several aspects mentioned in the main text. We unpack the details of the ***Omni6D*** dataset, exploring its structure and statistics. We provide the construction details of the latest datasets, ***Omni6D-xl*** and ***Omni6D-Real***. We provide a meticulous examination of the experimental procedures and analysis integral to our study. Additionally, we provided detailed insights into the questionnaire setting and result details regarding the visual realism of our Omni6D dataset. These supplemental details are invaluable in facilitating a better understanding of our research methods and discoveries.

B Dataset Details

B.1 Omni6D overview

Dataset structure. Our dataset is stored in folder-based structure. As illustrated in Fig. S1, it comprises symmetry annotations, point clouds sampled from 3D scanned objects with adjusted canonical poses, and rendered views. We also provide a Blender-based simulation framework to facilitate users.

Specifically for depth images, we applied a mapping transformation as mentioned in the main text. Original depth maps, saved as EXR files, have float32 precision with an accuracy of approximately $1e^{-7}$ and a size of 32 bits per pixel. Converting these depth maps to RGB format with a scaling factor of 10000 maintains a precision of about $1e^{-4}$, reducing storage size by 25% with 24 bits per pixel. Due to PNG compression, actual storage can be reduced to 5%-10% of the original size. Also, our depth map compression method enables direct visualization in PNG format.



```
Omni6D
├── info
│   ├── sym_info.csv
│   │   // rotational invariance of each mesh
│   ├── <train/val/test/test_unseen>_list.txt
│   │   // valid data list of each split
├── shape_data
│   ├── camera_<train/val/test/test_unseen>.pkl
│   │   // sampled 1024 points from each mesh,
│   │   // normalized to NOCS
│   ├── ShapeNetCore_<2048/4096>.h5
│   ├── ShapeNetCore_unseen_<2048/4096>.h5
│   │   // sampled 2048/4096 points from each mesh
├── CAMERA
│   ├── <train/val/test/test_unseen>
│   │   ├── <scene_id>
│   │   │   ├── <render_id>_color.png
│   │   │   │   // RGB image
│   │   │   ├── <render_id>_depth.png
│   │   │   │   // depth map
│   │   │   ├── <render_id>_coord.png
│   │   │   │   // NOCS mapping
│   │   │   ├── <render_id>_mask.png
│   │   │   │   // instance mask
│   │   │   ├── <render_id>_label.pkl
│   │   │   │   // ground truth annotations
```

Fig. S1: Dataset structure.

Table R1: Detailed statistical overview of Omni6D dataset. The table provides information about the number of categories, instances, and images in Omni6D$_{train}$, Omni6D$_{val}$, Omni6D$_{test}$ and Omni6D$_{out}$.

Datasets	# Categories	# Instances	# Images
Train	166	3,294	812,602
Val	166	919	28,661
Test	166	475	14,267
Out	17	52	4,762

Omni6D splits. Tab. R1 provides information about the number of categories, instances, and images in Omni6D$_{train}$, Omni6D$_{val}$, Omni6D$_{test}$ and Omni6D$_{out}$. The categories are shared amongst the training, validation, and testing datasets, with a distribution ratio of 7:2:1 for instances. On the other hand, Omni6D$_{out}$ stands distinct, comprising an added set of 17 categories. Each split's images are exclusively derived from its corresponding instances, yet all splits share rendering parameters and backgrounds uniformly. To enable comprehensive model training, we have augmented the training set with an extensive volume of rendered images, reaching a total of 0.8M.

Coordinate system. We formulate a unified 3D coordinate system for all pose labels, positioning the camera center as the origin. In relation to the image captured, we set +x to face outward, +y to point upwards, and +z towards the left. The pose of an object is recorded relative to what we term a canonical pose object. As illustrated in Fig. S2, an instance adjusted to the canonical pose has its bottom-face normal aligned with -y and its front-face aimed at +x(akin to being upright and facing forward). The camera's intrinsic parameters are established as [577.5, 577.5, 319.5, 239.5], with the image size defined as 640 x 480 pixels. All data attributes, including details concerning the object's position and dimensions, are denoted in metric units.

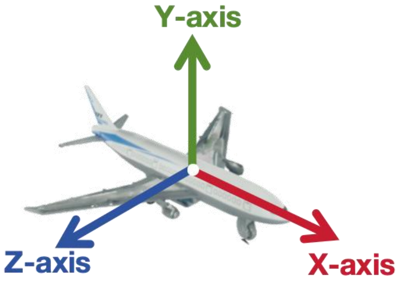


Fig. S2: An example instance adjusted to the canonical pose. The canonical plane has its bottom-face normal aligned with -y and its front-face aimed at +x(akin to being upright and facing forward).

Diversity of scenes. Each room is allocated a cube-shaped region, where objects are randomly positioned and fall free within room boundaries. Additionally, a lighting intensity range with a width of 2000 is established for each room model.

B.2 Omni6D Statistics

We first provide a category inventory and corresponding instance counts for each category within Omni6D in Fig. S9a. Most categories have [10, 50] objects.

In Section 4.1 of the main text, we mention cls$n$. Detailed categories from cls3 to cls48 are listed in Fig. S3. While subdividing the categories, we first select three categories that coincide with NOCS dataset [11], particularly those included in cls3: *bottle*, *bowl*, and *cup*. Then, for cls6, we opt for three categories similar in shape to those in cls3, namely *medicine_ bottle*, *shampoo*, and *red_ wine_ glass*. This selection aids in effectively finetuning the model across different categories. Following that, we generally select the remaining 42 categories based on the number of instances in each category, choosing from those with more instances to those with fewer.

B.3 Omni6D$_{\mathbf{out}}$ Statistics

In Section 4.3 of the main text, we undertake 6D object pose estimation studies on Omni6D$_{out}$. This process begins by loading the pre-trained Word2Vec model *GoogleNews-vectors-negative300.bin*. From the 166 categories available in Omni6D, we select the category that exhibits the highest cosine similarity with the unseen category for matching. As illustrated in Fig. S4, the text to the right of the bar graph clarifies which categories are ultimately matched with the unseen category displayed on the left. For each unseen category, our model presumes its category as the one that is matched and proceeds with pose estimation accordingly. This visual representation provides an intuitive understanding

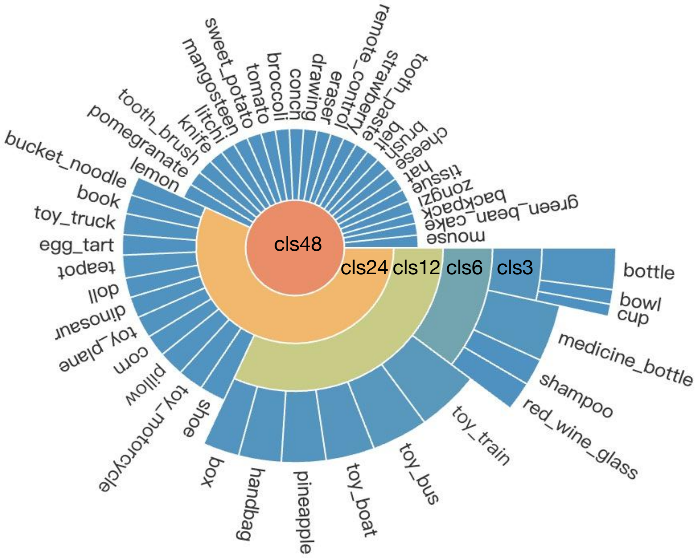


Fig. S3: Category inventory of cls$n$ within Omni6D. The angle of each sector in the chart reflects the relative size of the instance count within that category.

of how our model leverages this matching information to predict the pose for each unseen category. Likewise, when evaluating the unseen categories, we also annotated the symmetrical information and implemented the metric processing as outlined in Section 4.2.

C Omni6D-xl

Omni6D-xl extends Omni6D dataset by adding more categories and instance object models. Unlike normalizing all objects to the same scale, we retain the original scale of the objects and restore them to their actual size during rendering, adjusting other parameters accordingly. Moreover, we split our background rooms into training, validation, and test sets in a 2:1:1 ratio to avoid over-fitting on those scenes.

Dataset Collection. As shown in Tab. R2, Omni6D-xl comprises 13,161 instances across an impressive span of 339 categories. Each instance is a high-resolution textured mesh, obtained using Shining 3D scanner[1] and Artec Eva 3D scanner[2], collected from OmniObject3D [12]. We normalize object models to fit within a $(-1, 1)^3 (m^3)$ three-dimensional space, and align objects within each category to a consistent canonical pose. Additionally, we store the scale of the object models.

[1] https://www.einscan.com/

[2] https://www.artec3d.cn/

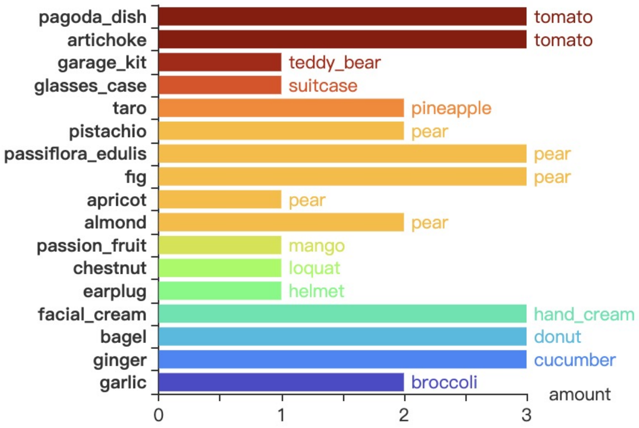


Fig. S4: Matching unseen categories from Omni6D$_{out}$ to Omni6D. The unseen categories from Omni6D$_{out}$ are listed on the left side of the bar graph, while the matched known categories from Omni6D are displayed on the right, clearly illustrating the optimal correspondence between unseen and known categories based on cosine similarity. The horizontal axis displays the instance count for each corresponding category. Bars of the same color underscore the same match.

Rendering. We employ stratified sampling to split instances within each category, subsequently dividing them into training, validation, and test sets in a 8:1:1 ratio. In constructing our dataset, we utilize 8 room models from the Replica dataset as backdrops, splitting them into training, validation, and test sets in a 2:1:1 ratio. For each scenery setup, we randomly select a room model to act as the background, along with 4-6 object instance models. Each room is allocated a cube-shaped region where objects are randomly positioned and allowed to fall freely within room boundaries, resulting in random scattering in a specific section of the room. Additionally, a lighting intensity range with a width of 2000 is established for each room model. Each object model is scaled by the pre-stored scale factor divided by 50. Considering the attention center of the combined instance models as the origin point, the camera randomly selects ten positions within an elevation angle range between $30 - 90°$. The camera then performs rendering at these selected positions while facing towards the attention center.
Setting. We utilize BlenderProc 2.5.0 [4] to implement the aforementioned rendering process. The intrinsic parameters of the camera are set to [577.5, 577.5, 319.5, 239.5], with an image size specified as 640×480. Our approach ensures the diversity and breadth of the dataset, making it suitable for rigorous testing and yielding accurate results.

D Omni6D-Real

To further validate the sim2real capability of models trained with Omni6D and reduce the gap between our dataset and real-world data, we constructed

Table R2: Comparisons between Omni6D, Omni6D-xl, Omni6D-Real and existing datasets. Our datasets significantly extend the range of everyday object categories and instances.

Datasets	Mode	Realism	# Categories	# Instances	# Images
ShapeNet-SRN Cars [9]	RGB	Synthetic	1	3514	-
Sim2Real Cars [9]	RGB	Real	1	10	-
CAMERA [11]	RGBD	Synthetic	6	1085	0.3M
REAL [11]	RGBD	Real	6	42	8k
Wild6D [13]	RGBD	Real	5	1722	1M
Omni6D-Real	RGBD	Real	39	73	1k
Omni6D	RGBD	Real-Scanned	166	4,688	0.8M
Omni6D-xl	RGBD	Real-Scanned	**339**	**13,161**	1.1M

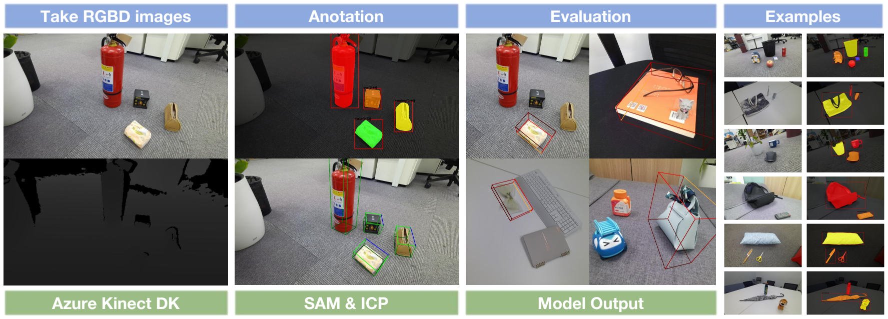


Fig. S5: Constructing Omni6D-Real: pipeline & examples.

a real-world dataset, ***Omni6D-Real***. As shown in Tab. R2, it comprises 30 scenes, 39 categories, 73 instances, and 1k images.

Dataset Construction. As shown in Fig. S5, we captured RGBD images with the Azure Kinect DK[3] and preprocessed them using SAM [6] for object masks and ICP [2] for point cloud registration. The intrinsic parameters of the camera are set to [605.81, 605.63, 641.72, 363.23], with an image size specified as 1280×720. For each scene, we manually annotated 3D bounding boxes for the first frame and derived bboxes for the next frame based on registered poses. Addressing the inherent limitations of ICP, particularly its accumulating errors, we further refined the derived bboxes through manual adjustments. This iterative process, where ICP serves as an aid to manual annotation, ensures the accuracy of 3D bboxes across all frames.

Evaluation. We evaluated the performance of DualPoseNet [7] on our processed real-world dataset. Despite being trained solely on simulated data, the model exhibited excellent performance on real-world tasks. This demonstrates to a certain extent that our real-scanned 3D models can minimize the gap between synthetic and real images.

[3] https://learn.microsoft.com/azure/kinect-dk/

Table R3: Detailed parameters. Experimental settings on different baselines.

Model	Learning_rate	Batch_size	# GPUs
SPD [10]	1e-4	128	4
SGPA [3]	1e-4	128	4
DualPoseNet [7]	1e-4	128	1
RBP-Pose [14]	1e-4	256	4
GPV-Pose [5]	1e-4	256	4
HS-Pose [15]	1e-4	256	4

Table R4: Performance of top-20 categories on Omni6D. Models are trained on Omni6D$_{train}$ and tested on Omni6D$_{test}$. The table demonstrates the average performance of each algorithm across the top 20 categories, as measured by the $5°2cm$ metric. **Bold** and underlined results indicate the best and second-best performers.

Methods	Network	IoU_{50}	IoU_{75}	$5°2cm$	$5°5cm$	$10°2cm$	$10°5cm$	$5°$	$10°$	$2cm$	$5cm$
SPD [10]	implicit	68.65	45.27	**24.19**	**26.78**	37.18	**42.15**	**27.60**	**43.34**	60.49	84.03
SGPA [3]	implicit	70.40	**48.23**	20.17	21.79	**37.30**	40.78	22.26	41.87	63.39	86.14
DualPoseNet [7]	hybrid	**74.09**	41.50	15.56	17.11	30.25	32.78	17.14	32.84	83.48	**98.46**
RBP-Pose [14]	hybrid	42.03	10.74	2.84	4.54	3.41	5.21	4.68	5.37	44.77	88.43
GPV-Pose [5]	explicit	19.53	0.78	0.74	3.17	0.81	3.58	7.06	7.85	7.03	39.86
HS-Pose [15]	explicit	72.36	37.81	11.94	13.26	22.08	23.93	13.37	24.07	**86.11**	98.31

E Additional Experimental Details

E.1 Experimental Settings

All experiments are conducted on a server equipped with 96 Intel(R) Xeon(R) Gold 6248R CPUs @ 3.00GHz and 8 NVIDIA A100-SXM4-80GB GPUs. We ensure consistency in all parameters and strategies throughout training, thereby maintaining uniformity in our experimental environment. For our baseline model, we adhere to the same parameters as provided by the original authors, with modifications only made to *learning_rate*, *batch_size*, and the corresponding number of GPUs used. Detailed parameters are displayed in Tab. R3.

We encountered some challenges during model training. Due to the larger batch size we selected compared to the original model, the training speed of the GPV-Pose model became excessively slow. The main reason for this issue is that GPV-Pose [5] model uses "for loop" for batch processing during training, which is inefficient when dealing with large-scale data. We optimized the model by replacing "for loop" with batch computations carried out at the Tensor level. This modification significantly accelerated our training speed, effectively ensuring the efficient functioning of the model.

E.2 Performance on Omni6D

In this section, we provide the results of the 5° and 2 *cm* metrics for categories in Omni6D. Fig. S6 showcases the 5°(R5) and 2 *cm*(T2) metrics for various

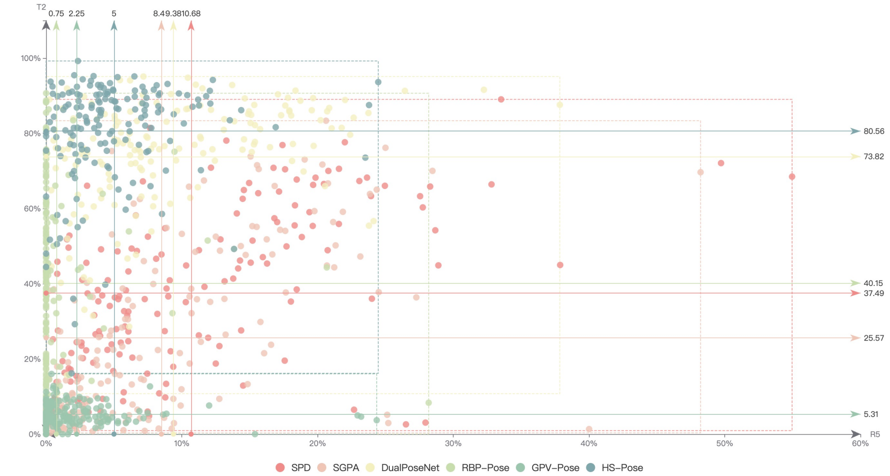


Fig. S6: Metrics 5° and 2 *cm* results on Omni6D categories. It showcases the 5° (R5) and 2 *cm* (T2) metrics for various models across different categories on the Omni6D test set. Each color represents a model, with each point indicating a category result. Dashed lines outline the range of each model's 5° (R5) and 2 *cm* (T2) metrics, while arrows depict their means.

models across different categories on the Omni6D test set. The results show that SPD and SGPA excel particularly in predicting rotations, potentially due to their implicit networks' tendency to generate more accurate rotational predictions. On the other hand, DualPoseNet, HS-Pose and RBP-Pose offer superior estimates for translations, likely related to the capabilities of explicit network models to deliver better translation and size estimations. These findings further affirm the speculations made in Section 4.3.

Tab. R4 demonstrates the average performance of each algorithm across the top 20 categories, as measured by the 5°2 *cm* metric. As shown in the table, it's evident that all algorithms show improved performance across various metrics compared to the full set of 166 categories, which is foreseeable. While all algorithms see similar improvements, SPD and SGPA stand out with notable progress. Considering their bad performance on unseen categories, as outlined in the main text, it's clear that they exhibit considerable variability in predictive accuracy across different categories. This suggests that SPD and SGPA employ a nuanced approach, finetuning their strategies for each category by leveraging their implicit network methodologies. These methodologies sync well with specific features and challenges of certain categories, enabling more accurate predictions. Conversely, their effectiveness lessens when applied to categories that mismatch their methodologies.

We also report the non-symmetry-aware metric results in Tab. R5, showing a notable performance drop compared to the symmetry-aware metric presented

Table R5: Non-symmetry-aware metric results on Omni6D. Models are trained on Omni6D$_{train}$ and tested on Omni6D$_{test}$, while not using our symmetry-aware metric.

Methods	IoU_{50}	IoU_{75}	$5°2cm$	$5°5cm$	$10°2cm$	$10°5cm$	$5°$	$10°$	$2cm$	$5cm$
SPD [10]	30.82	**13.09**	**3.36**	**3.62**	**8.10**	**9.06**	**3.65**	**9.19**	38.32	71.43
SGPA [3]	26.43	10.06	2.34	2.57	6.25	7.40	2.59	7.62	26.11	60.67
DualPoseNet [7]	35.78	12.32	2.06	2.11	6.47	6.74	2.11	6.75	74.13	96.42
RBP-Pose [14]	14.77	0.63	0.00	0.00	0.00	0.01	0.00	0.01	34.33	73.54
GPV-Pose [5]	5.50	0.02	0.00	0.01	0.01	0.04	0.02	0.07	5.37	33.31
HS-Pose [15]	**39.18**	9.68	0.36	0.37	2.30	2.43	0.37	2.44	**80.65**	**97.64**

Table R6: Individual category performance on unseen categories. Models are trained on Omni6D$_{train}$ and tested on Omni6D$_{out}$, using the optimal DualPoseNet [7] model. The table distinctly presents results for each category, with the 1st column representing the category name and the 2nd column indicating the corresponding known matched category. The table is sorted in descending order based on the metric $5°2cm$.

Category	Match	IoU_{50}	IoU_{75}	$5°2cm$	$5°5cm$	$10°2cm$	$10°5cm$	$5°$	$10°$	$2cm$	$5cm$
passion_fruit	mango	58.79	25.08	8.44	8.77	18.51	18.99	8.77	19.48	85.23	99.19
facial_cream	hand_cream	53.43	28.61	7.58	7.82	16.38	17.60	7.82	17.60	84.11	96.82
taro	pineapple	58.70	26.48	5.50	5.65	16.49	16.79	5.65	16.79	89.47	99.39
fig	pear	60.64	24.79	3.52	3.63	15.69	15.90	3.95	16.33	84.85	99.68
garlic	broccoli	32.85	4.78	3.18	3.18	6.16	6.58	3.18	6.79	81.95	99.79
earplug	helmet	35.04	14.14	2.69	3.08	9.23	9.87	3.08	9.87	88.72	98.72
passiflora_edulis	pear	37.76	15.22	1.82	1.82	7.29	7.75	1.82	7.75	77.51	99.24
bagel	donut	38.71	13.92	1.75	2.25	9.64	10.26	2.25	10.26	83.35	99.12
artichoke	tomato	48.20	13.77	0.78	0.90	3.70	4.26	1.01	4.48	79.28	99.22
pagoda_dish	tomato	32.66	4.45	0.78	0.91	2.22	2.61	0.91	3.00	82.40	98.31
ginger	cucumber	30.33	1.54	0.78	0.78	2.08	2.86	0.78	2.86	59.48	98.70
almond	pear	23.43	1.88	0.76	0.89	1.78	2.28	1.02	2.41	80.58	99.11
garage_kit	teddy_bear	26.66	5.44	0.59	0.73	2.42	2.87	0.83	3.08	71.59	97.09
glasses_case	suitcase	37.99	5.24	0.44	0.44	0.88	1.32	0.44	1.32	90.79	96.93
chestnut	loquat	22.73	1.97	0.27	0.40	0.93	1.19	0.40	1.19	57.43	97.75
pistachio	pear	23.56	2.03	0.17	0.17	0.34	0.52	0.17	0.69	80.72	99.66
apricot	pear	11.48	0.26	0.00	0.00	0.00	0.00	0.00	0.00	80.32	100.00

in Tab. 2. As discussed in Fig. 2, the prevalence of rotational invariance in 3D models makes the consideration of symmetry indispensable.

E.3 Generalization Performance

Tab. R6 distinctly presents the results for each category, derived from tests using the optimal DualPoseNet [7] model. In this table, the first column lists the category name while the second column indicates the corresponding known matched category. It can be observed that prediction for translation is almost category-independent, while rotation is closely related to the category.

Table R7: Performance of SPD on Omni6D dataset trained from scratch. It presents the performance of the SPD model when trained from scratch separately on various subsets of the Omni6D dataset, specifically cls3, cls6, cls12, cls24, and cls48, each of which contains a different number of categories.

Train	Test	IoU_{50}	IoU_{75}	$5^\circ 2cm$	$5^\circ 5cm$	$10^\circ 2cm$	$10^\circ 5cm$	5°	10°	$2cm$	$5cm$
train-from-scratch(cls3)	cls3	44.27	20.50	9.52	10.56	14.45	17.13	10.85	17.98	41.98	65.42
train-from-scratch(cls6)	cls6	54.94	28.37	14.96	16.86	20.75	24.91	17.26	25.62	51.13	74.89
train-from-scratch(cls12)	cls12	55.30	29.47	12.92	15.01	21.90	26.31	15.59	27.58	49.99	77.51
train-from-scratch(cls24)	cls24	57.37	31.08	11.90	13.44	22.67	26.36	14.02	27.62	52.98	79.73
train-from-scratch(cls48)	cls48	48.22	24.54	9.07	11.07	17.53	22.15	11.89	23.63	41.60	73.28

Table R8: Performance of SPD on Omni6D dataset with finetuning strategy. It presents the performance of the SPD model initially pretrained on CAMERA dataset [11] and then incrementally finetuned using various subsets of the Omni6D dataset, specifically cls3, cls6, cls12, cls24, and cls48.

Train	Test	IoU_{50}	IoU_{75}	$5^\circ 2cm$	$5^\circ 5cm$	$10^\circ 2cm$	$10^\circ 5cm$	5°	10°	$2cm$	$5cm$
pretrain (CAMERA)	cls3	16.63	0.79	0.05	0.59	0.09	0.93	2.55	3.60	2.29	23.53
finetune (CAMERA+cls3)	cls3	46.19	21.42	10.20	11.49	16.71	19.59	11.79	20.18	53.39	79.83
finetune (CAMERA+cls6)	cls6	60.85	32.57	15.09	17.84	23.63	28.54	18.02	28.82	62.89	86.03
finetune (CAMERA+cls12)	cls12	56.67	29.71	13.18	15.08	22.54	26.34	15.50	26.92	58.31	83.68
finetune (CAMERA+cls24)	cls24	55.82	28.81	12.06	13.58	22.24	26.02	14.03	26.95	57.92	84.36
finetune (CAMERA+cls48)	cls48	45.06	22.76	9.22	11.22	17.10	21.36	11.96	22.67	44.80	74.66

E.4 Category-wise Analysis

In the corresponding subsection under Section 4.3, we introduce the concept of diversity. Assume that C_i is the set of all instances within category i, c_{ij} and c_{ik} are two instances within this set, and $\text{Chamfer}(c_{ij}, c_{ik})$ is the Chamfer distance [1] between instances c_{ij} and c_{ik}. Then, the diversity D_i within category i can be calculated as:

$$D_i = \frac{1}{|C_i|^2} \sum_{j=1}^{|C_i|} \sum_{k=1}^{|C_i|} \text{Chamfer}(c_{ij}, c_{ik}). \tag{1}$$

Essentially, this formula calculates the average Chamfer distance among all possible pairs of instances within a category, serving as a measure of diversity for that category. A larger result indicates higher intra-class diversity among instances within that category. Fig. S9b depicts the intra-class diversity across various categories in Omni6D.

E.5 Finetune from Limited Categories

As elaborated in the corresponding subsection under Section 4.3 in the main text, Tabs. R7 to R12 respectively present the specific numerical results of the

Table R9: Performance of DualPoseNet on Omni6D trained from scratch. It presents the performance of the DualPoseNet model when trained from scratch separately on various subsets of Omni6D.

Train	Test	IoU_{50}	IoU_{75}	$5°2cm$	$5°5cm$	$10°2cm$	$10°5cm$	$5°$	$10°$	$2cm$	$5cm$
train-from-scratch(cls3)	cls3	66.53	39.60	17.03	17.24	29.20	29.68	17.24	29.68	90.29	96.60
train-from-scratch(cls6)	cls6	76.27	44.62	20.59	21.12	32.59	33.84	21.16	33.90	87.75	96.81
train-from-scratch(cls12)	cls12	68.21	37.83	17.06	18.02	27.86	29.70	18.08	29.79	81.73	96.52
train-from-scratch(cls24)	cls24	70.14	43.01	19.99	20.92	33.03	34.83	21.03	34.94	82.58	96.90
train-from-scratch(cls48)	cls48	65.00	33.47	10.18	11.07	23.43	25.50	11.12	25.63	76.38	96.48

Table R10: Performance of DualPoseNet on Omni6D with finetuning strategy. It presents the performance of the DualPoseNet model initially pretrained on CAMERA dataset [11] and then incrementally finetuned using various subsets of Omni6D.

Train	Test	IoU_{50}	IoU_{75}	$5°2cm$	$5°5cm$	$10°2cm$	$10°5cm$	$5°$	$10°$	$2cm$	$5cm$
pretrain (CAMERA)	cls3	29.04	5.45	3.42	3.91	3.92	4.48	4.05	4.62	67.28	89.58
finetune (CAMERA+cls3)	cls3	75.25	44.57	17.72	17.72	32.95	33.70	17.72	33.70	91.52	96.51
finetune (CAMERA+cls6)	cls6	77.34	46.32	23.17	23.66	34.00	35.27	23.73	35.37	89.96	97.32
finetune (CAMERA+cls12)	cls12	68.61	37.58	17.17	17.88	28.15	29.45	17.94	29.58	83.22	96.83
finetune (CAMERA+cls24)	cls24	70.68	43.00	22.55	22.96	33.82	35.61	22.96	35.61	89.62	96.83
finetune (CAMERA+cls48)	cls48	64.60	34.52	13.57	14.36	25.34	27.01	14.45	27.20	77.68	96.08

training from scratch and finetuning experiments conducted by SPD, DualPoseNet, and HS-Pose.

For the training from scratch experiments, it is observed that an increase in the number of categories during the training and testing phases generally leads to a decline in most performance indicators. Contrastingly, in the finetuning experiments, as the number of categories used for finetuning and testing increases, most performance indicators do show a decline. However, certain metrics like 5 cm remain relatively stable, and the decrease in other metrics isn't as severe as when training from scratch. This observation points to the robustness of the pretraining and incremental finetuning approach across a different number of categories, emphasizing its effectiveness.

E.6 Qualitative Comparisons

For category-level 6D pose and size estimation, we visualize more qualitative results of different methods on Omni6D$_{test}$ and Omni6D$_{out}$ in Fig. S10 and Fig. S11. These figures illustrate the models' ability to generalize within known categories (intra-class generalization) as well as across unseen categories (inter-class generalization).

Table R11: Performance of HS-Pose on Omni6D trained from scratch. It presents the performance of the HS-Pose model when trained from scratch separately on various subsets of Omni6D.

Train	Test	IoU_{50}	IoU_{75}	$5^\circ 2cm$	$5^\circ 5cm$	$10^\circ 2cm$	$10^\circ 5cm$	5°	10°	$2cm$	$5cm$
train-from-scratch(cls3)	cls3	94.46	86.57	47.65	48.28	81.81	83.70	48.54	83.96	90.33	97.21
train-from-scratch(cls6)	cls6	93.61	82.65	48.79	49.93	74.03	76.24	49.93	76.33	90.71	97.68
train-from-scratch(cls12)	cls12	81.40	57.79	21.78	22.13	42.79	43.93	22.13	43.94	87.48	98.45
train-from-scratch(cls24)	cls24	79.75	52.25	16.92	17.58	37.17	38.93	17.59	38.95	87.66	98.38
train-from-scratch(cls48)	cls48	73.30	39.62	8.79	9.16	23.97	25.41	9.18	25.49	83.69	98.42

Table R12: Performance of HS-Pose on Omni6D with finetuning strategy. It presents the performance of the HS-Pose model initially pretrained on CAMERA dataset [11] and then incrementally finetuned using various subsets of Omni6D.

Train	Test	IoU_{50}	IoU_{75}	$5^\circ 2cm$	$5^\circ 5cm$	$10^\circ 2cm$	$10^\circ 5cm$	5°	10°	$2cm$	$5cm$
pretrain (CAMERA)	cls3	31.67	7.14	4.13	5.12	6.38	7.91	5.19	8.25	70.56	92.27
finetune (CAMERA+cls3)	cls3	94.04	88.29	62.52	63.92	84.51	87.41	63.92	87.41	90.87	97.60
finetune (CAMERA+cls6)	cls6	94.47	86.19	56.20	58.06	79.82	82.76	58.10	82.89	90.76	97.87
finetune (CAMERA+cls12)	cls12	83.85	61.37	28.37	29.03	48.73	50.37	29.03	50.43	87.48	97.98
finetune (CAMERA+cls24)	cls24	81.35	56.59	23.27	24.04	43.22	45.16	24.05	45.22	87.68	98.43
finetune (CAMERA+cls48)	cls48	75.18	45.11	14.42	15.02	30.45	32.24	15.02	32.31	83.34	98.45

F Visual Realism

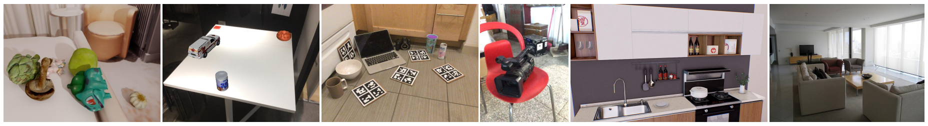
F.1 Questionnaire settings

We evaluated the visual realism of Omni6D in comparison to other datasets through a survey involving 70 human subjects. We randomly selected 10 images from Omni6D, CAMERA [11], REAL [11], and Wild6D datasets [13]. To introduce noise, we blended in 2 images from COCO [8], which includes captured photos, and 3 images from SKETCH[4], which comprises rendered images. We randomly shuffled the order of the aforementioned 45 images and asked subjects to rate them anonymously, *i.e.*, participants were unaware of the dataset to which each image belonged. Subjects were asked to rate the realism of sampled images on a scale from 1 (least realistic) to 5 (most realistic). Here is the specific instruction for this survey: *In this subsection, participants are required to rate the fidelity of the images, i.e., how closely they resemble images seen by the human eye. Ratings range from 1 to 5, with 1 representing a complete absence of fidelity and 5 denoting full congruence with perceptual images.*

F.2 Questionnaire results

We reported the average ratings and standard deviations for all datasets in Fig. S7, along with a sampled image from the questionnaire. Fig. S8 illustrates

[4] https://sketchfab.com/



Omni6D
2.69 ± 0.39

CAMERA
1.55 ± 0.08

REAL
3.53 ± 0.28

WILD6D
4.38 ± 0.23

SKETCH
2.42 ± 0.23

COCO
3.48 ± 0.64

Fig. S7: Comparison of Visual Realism. Complete results, including ratings for all datasets in the survey.

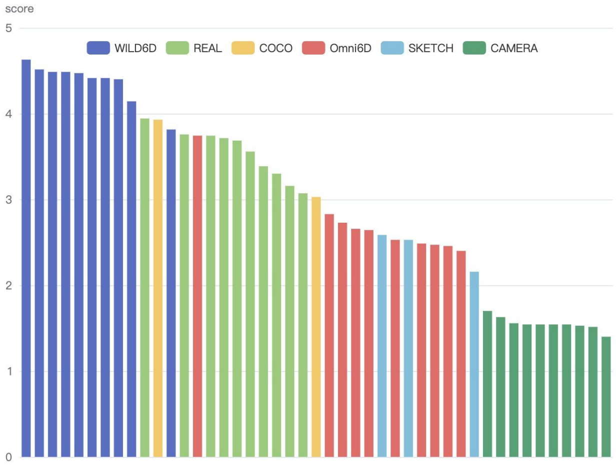
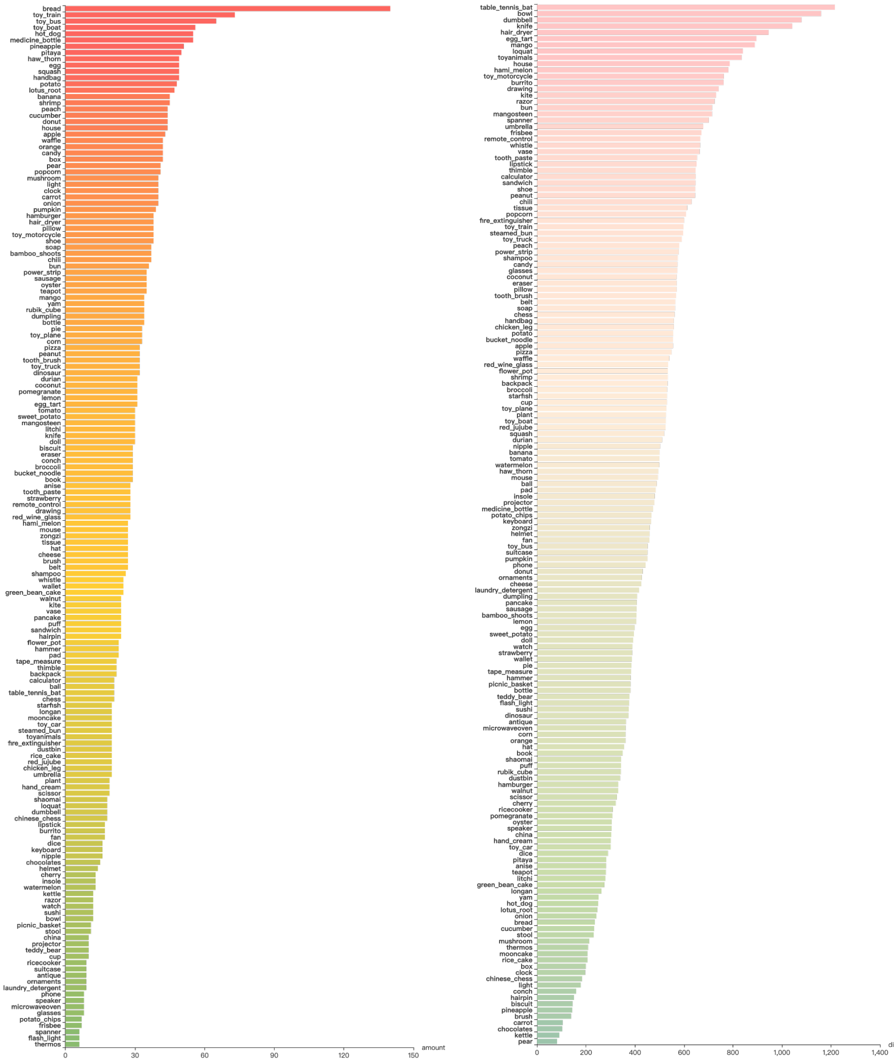


Fig. S8: Fidelity ratings for each image. It displays the average ratings of all images in the questionnaire across 70 surveys, while the bar chart shows a gradual decrease in ratings from left to right, with each color representing a different dataset.

the average rating for each image. It can be observed that despite Omni6D having lower fidelity compared to captured photos, its ratings are significantly higher than those of CAMERA, which are also synthetic images. Furthermore, there is a noticeable gap between the ratings of Omni6D and CAMERA, with some images from Omni6D closely resembling captured photos.

References

1. Barrow, H.G., Tenenbaum, J.M., Bolles, R.C., Wolf, H.C.: Parametric correspondence and chamfer matching: Two new techniques for image matching. In: IJCAI. pp. 659–663. William Kaufmann (1977)
2. Besl, P.J., McKay, N.D.: A method for registration of 3-d shapes. IEEE Trans. Pattern Anal. Mach. Intell. **14**(2), 239–256 (1992)
3. Chen, K., Dou, Q.: SGPA: structure-guided prior adaptation for category-level 6d object pose estimation. In: ICCV. pp. 2753–2762 (2021)
4. Denninger, M., Winkelbauer, D., Sundermeyer, M., Boerdijk, W., Knauer, M., Strobl, K.H., Humt, M., Triebel, R.: Blenderproc2: A procedural pipeline for photorealistic rendering. J. Open Source Softw. **8**(83), 4901 (2023)
5. Di, Y., Zhang, R., Lou, Z., Manhardt, F., Ji, X., Navab, N., Tombari, F.: Gpv-pose: Category-level object pose estimation via geometry-guided point-wise voting. In: CVPR. pp. 6771–6781 (2022)
6. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything. arXiv:2304.02643 (2023)
7. Lin, J., Wei, Z., Li, Z., Xu, S., Jia, K., Li, Y.: Dualposenet: Category-level 6d object pose and size estimation using dual pose network with refined learning of pose consistency. In: ICCV. pp. 3540–3549 (2021)
8. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. pp. 740–755 (2014)
9. Lin, Y., Florence, P., Barron, J.T., Rodriguez, A., Isola, P., Lin, T.: inerf: Inverting neural radiance fields for pose estimation. In: IROS. pp. 1323–1330 (2021)
10. Tian, M., Ang, M.H., Lee, G.H.: Shape prior deformation for categorical 6d object pose and size estimation. In: ECCV (21). pp. 530–546 (2020)
11. Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., Guibas, L.J.: Normalized object coordinate space for category-level 6d object pose and size estimation. In: CVPR. pp. 2642–2651 (2019)
12. Wu, T., Zhang, J., Fu, X., Wang, Y., Ren, J., Pan, L., Wu, W., Yang, L., Wang, J., Qian, C., Lin, D., Liu, Z.: Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In: CVPR. pp. 803–814 (2023)
13. Ze, Y., Wang, X.: Category-level 6d object pose estimation in the wild: A semi-supervised learning approach and A new dataset. In: NeurIPS (2022)
14. Zhang, R., Di, Y., Lou, Z., Manhardt, F., Tombari, F., Ji, X.: Rbp-pose: Residual bounding box projection for category-level pose estimation. In: ECCV (1). pp. 655–672 (2022)
15. Zheng, L., Wang, C., Sun, Y., Dasgupta, E., Chen, H., Leonardis, A., Zhang, W., Chang, H.J.: Hs-pose: Hybrid scope feature extraction for category-level object pose estimation. In: CVPR. pp. 17163–17173 (2023)



(a) Instance count of category

(b) Intra-class diversity of category

Fig. S9: Omni6D Statistics. (a) Category inventory and instance counts within Omni6D. Bars are sorted in descending order based on the instance counts of each category in the entire Omni6D dataset (train/val/test). **(b)** Intra-class diversity within categories in Omni6D. We measure the diversity of instances within a category using the mean Chamfer distance of all pairwise pairs within that category. Bars are sorted in descending order based on the intra-class diversity of each category in Omni6D$_{train}$.

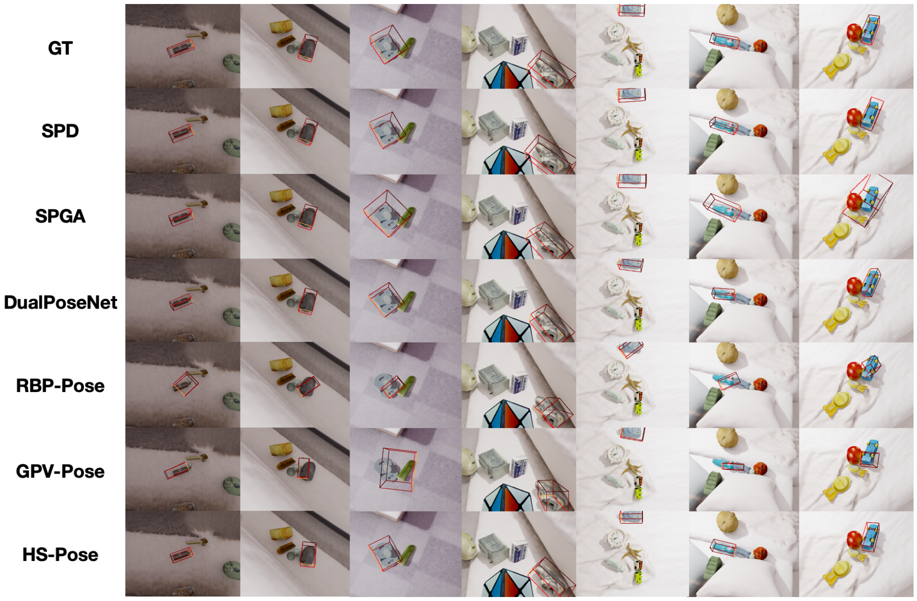


Fig. S10: Qualitative 6D pose and size estimation on Omni6D. From top to bottom, figures correspond to results of ground truth, SPD [10], SGPA [3], DualPoseNet [7], RBP-Pose [14], GPV-Pose [5], HS-Pose [15] on Omni6D$_{test}$.

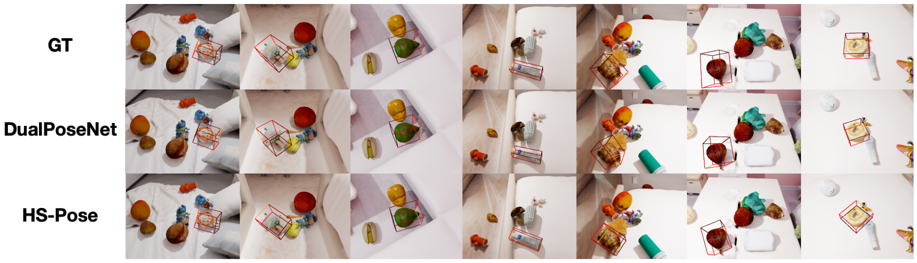


Fig. S11: Qualitative 6D pose and size estimation on unseen categories. From top to bottom, figures correspond to results of ground truth, DualPoseNet [7] and HS-Pose [15] on Omni6D$_{out}$. We only showcase results from two models, DualPoseNet and HS-Pose, both of which exhibit inter-class generalization abilities.