LEMON: Learning 3D Human-Object Interaction Relation from 2D Images

Supplementary Material

Contents

A. Implementation Details

In this section, we clarify details about the methods, experiments, and training settings.

A.1. Method Details

We provide a table to display dimensions and meanings of tensors in the LEMON pipeline, shown in Tab. 1. The input images are resized to 224×224 and a pre-trained HRNet is taken as the extractor. We add human and object masks in the training to enable the model to focus efficiently on the interaction between the human and object. The image extractor outputs the image feature with the shape $\mathbb{R}^{2048\times7\times7}$. A 1×1 convolutional layer is used to reduce the feature dimension and the feature is flattened to $\mathbf{F}_i \in \mathbb{R}^{768\times49}$. The number of object point clouds is 2048, the same setting as 3D-AffordanceNet [11] and IAG-Net [51]. For humans, the vertices of SMPL-H [42] are regarded as the input, raw vertices possess the shape $\mathbb{R}^{6890\times3}$. We sample it to $\mathbb{R}^{1723\times3}$ through the script in COMA [41], which is also utilized in BSTRO [19]. Note that we uniformly use N_h to represent the number of vertices in the main paper for simplification. Actually, N_h in H and $\bar{\phi}_c$ are 6890, while in other features are 1723. DGCNN is taken as the backbone network for extracting point-wise features of the human and object and output the $\mathbf{F}_o \in \mathbb{R}^{768\times2048}, \mathbf{F}_h \in \mathbb{R}^{768\times1723}$.

The tokens $\mathbf{T}_o, \mathbf{T}_h \in \mathbb{R}^{768\times1}$ are utilized to represent interaction intentions of geometries. They are concatenated with $\mathbf{F}_o, \mathbf{F}_h$, and get the feature sequence $\mathbf{F}_{to} \in \mathbb{R}^{768\times2049}, \mathbf{F}_{th} \in \mathbb{R}^{768\times1724}$. Then, the multi-branch attention f_δ is performed on $\mathbf{F}_i$ and $\mathbf{F}_{to}, \mathbf{F}_{th}$. $\mathbf{F}_i$ serves as the shared key and value, $\mathbf{F}_{to}, \mathbf{F}_{th}$ serve as queries in two branches. f_δ has 12 heads and each head with the dimension of 64. After f_δ, $\mathbf{F}_{to}, \mathbf{F}_{th}$ are updated to $\bar{\mathbf{F}}_{to}, \bar{\mathbf{F}}_{th}$ with the same shape, which are then split to $\bar{\mathbf{F}}_o \in \mathbb{R}^{768\times2048}$, $\bar{\mathbf{T}}_o \in \mathbb{R}^{768\times1}$ and $\bar{\mathbf{F}}_h \in \mathbb{R}^{768\times1723}$, $\bar{\mathbf{T}}_h \in \mathbb{R}^{768\times1}$.

Table 1. **Tensors.** The dimension and meaning of the tensors in the LEMON pipeline.

Tensor	Dimension	Meaning
$\mathbf{F}_i$	768×49	image feature
$\mathbf{F}_o, \bar{\mathbf{F}}_o$	768×2048	geometric feature of the object
$\mathbf{F}_h, \bar{\mathbf{F}}_h$	768×1723	geometric feature of the human
$\mathbf{T}_o, \bar{\mathbf{T}}_o$	768×1	intention tokens of object geometry
$\mathbf{T}_h, \bar{\mathbf{T}}_h$	768×1	intention tokens of human geometry
$\mathbf{F}_{to}$	768×2049	concatenation of $\mathbf{F}_o, \mathbf{T}_o$
$\bar{\mathbf{F}}_{to}$	768×2049	concatenation of $\bar{\mathbf{F}}_o, \bar{\mathbf{T}}_o$
$\mathbf{F}_{th}$	768×1724	concatenation of $\mathbf{F}_h, \mathbf{T}_h$
$\bar{\mathbf{F}}_{th}$	768×1724	concatenation of $\bar{\mathbf{F}}_h, \bar{\mathbf{T}}_h$
C_o	1×2048	curvature of the object geometry
C_h	1×1723	curvature of the human geometry
$\bar{C}_o$	768×2048	curvature feature of the object
$\bar{C}_h$	768×1723	curvature feature of the human
$\bar{\mathbf{F}}_{co}$	768×2048	geometric feature with curvature
$\bar{\mathbf{F}}_{ch}$	768×1723	geometric feature with curvature
$\mathbf{T}_{sp}$	768×3	spatial token sequence
ϕ_a	768×2048	object affordance representation
ϕ_c	768×1723	human contact representation
ϕ_p	768×3	object spatial representation

$\bar{\mathbf{F}}_o, \bar{\mathbf{F}}_h$ and $\bar{\mathbf{T}}_o, \bar{\mathbf{T}}_h$ are utilized to model the geometric correlation. Firstly, we introduce the method to calculate the geometric curvature. Shown in Fig. 1, for each point p in the point cloud, the neighbor points are utilized to estimate its normal curvature. Assume p has n neighbor points, and let m_i be the i-th neighbor point. The normal vector corresponding to m_i is $\vec{M}_i$. Define $p, \vec{X}_i, \vec{Y}_i, \vec{N}_i$ be an orthogonal coordinates system, which is the local coordinates L at point p. $\vec{N}_i$ is the normal vector of p, $\vec{X}_i, \vec{Y}_i$ are orthogonal unit vectors. In L, coordinates could be formulated as: $p(0,0,0), m_i(x_i, y_i, z_i), \vec{M}_i(n_{x,i}, n_{y,i}, n_{z,i})$. Then, the normal curvature C_n^i of p could be calculated with an osculating circle passing through point p and m_i, which could be expressed as:

$$C_n^i = -\frac{\sin\beta}{|pm_i|\sin\theta} \approx -\frac{n_{xy}}{\sqrt{n_{xy}^2+n_z^2}\sqrt{x_i^2+y_i^2}}, \quad where \quad n_{xy} = \frac{x_i n_{x,i} + y_i n_{y,i}}{\sqrt{x_i^2+y_i^2}}, n_z = n_{z,i}, \tag{1}$$

where θ is the included angle between vectors -$\vec{N}_i$ and $p\vec{m}_i$, β is between vectors $\vec{N}$ and $\vec{M}_i$. Taking this method to obtain the curvature C_o, C_h. For the convenience of future research, we store these curvatures for direct use by other researchers. As described in the main paper, C_o, C_h are encoded into high dimension, and the cross-attention f_m is mutually performed on them. f_m also possesses 12 heads

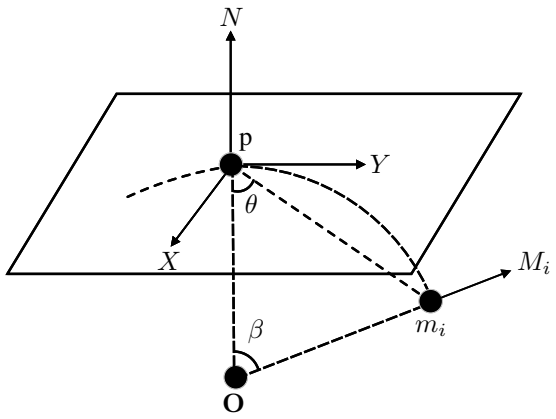

Figure 1. The local coordinates system L with the triangle defined by the osculating circle neighbor point of p.

and each head with the dimension of 64. The fusion layer f (Sec. 3.2 in the main paper) is the 1×1 convolution layer, which reduces the fused feature dimension to 768 for subsequent calculations.

To model the object's spatial representation, the token sequence $\mathbf{T}_{sp} \in \mathbb{R}^{768\times 3}$ is concatenated with the semantic token $\bar{\mathbf{T}}_o$ and the global geometric feature of the object. With the addition of positional encoding, the concatenate feature is utilized to query the corresponding feature of the human through a cross-attention layer f_ρ. The global geometric features are obtained by max-pooling $\bar{\mathbf{F}}_{co}, \bar{\mathbf{F}}_{ch}$, f_ρ has the same architecture with f_m.

The decoder has three heads to project the output of human contact $\bar{\phi}_c$, object affordance $\bar{\phi}_a$, and object center position $\bar{\phi}_p$. Each head is composed of a linear layer, a batch-normalization layer, and an activation layer. ϕ_a is projected to $\bar{\phi}_a \in \mathbb{R}^{2048\times 1}$, and ϕ_p is projected to $\bar{\phi}_p \in \mathbb{R}^3$. For the contact feature, it is first projected to $\mathbb{R}^{1723\times 1}$ and then up-project to $\bar{\phi}_c \in \mathbb{R}^{6890\times 1}$ by another linear layer. We give the formulation of $\mathcal{L}_p, \mathcal{L}_s$ in the main paper. Here, we also provide the formulation of $\mathcal{L}_c, \mathcal{L}_a$, which are the same, expressed as:

$$\begin{aligned}\mathcal{L}_c, \mathcal{L}_a =& 1 - \frac{\sum_j^N yx + \epsilon}{\sum_j^N y + x + \epsilon} - \frac{\sum_j^N (1-y)(1-x) + \epsilon}{\sum_j^N 2 - y - x + \epsilon} \\ &+ \frac{1}{N}\sum_j^N [-(1-\alpha)(1-y)\, x^\gamma \log(1-x) \\ &- \alpha y (1-x)^\gamma \log(x)],\end{aligned} \tag{2}$$

where N indicates the number of points within each geometry, x is the prediction, y is the ground truth, ϵ is set to $1e$-6, α, γ are set to 0.25 and 2 respectively.

A.2. Benchmark Details

Evaluation Metrics. We refer to methods that estimate each interaction element to benchmark the 3DIR [11, 19, 39, 45, 51]. Specifically, the Precision, Recall, F1 score, and geodesic distance are utilized to evaluate the contact estimation. AUC, aIOU, and SIM are utilized to evaluate the anticipation of object affordance. MSE is taken to evaluate the prediction of objects' spatial positions. The details of these metrics are as follows:

- **Precision, Recall, F1** [14]: Precision is the ratio of correctly predicted positive observations to the total predicted positives, measures the accuracy of the positive predictions made by a model. Recall is the ratio of correctly predicted positive observations to all observations in the actual class and measures the ability of a model to capture all the positive instances. F1-score is the harmonic mean of Precision and Recall. It provides a balance between Precision and Recall, making it a suitable metric when there is an imbalance between classes. They could be formulated as:

$$\begin{aligned} Precision &= \frac{TP}{TP+FP}, \\ Recall &= \frac{TP}{TP+FN}, \\ F1 &= \frac{2\cdot Precision \cdot Recall}{Precision + Recall}, \end{aligned} \tag{3}$$

  where TP, FP, and FN denote the true positive, false positive, and false negative counts, respectively.
- **geodesic distance** [19]: The geodesic distance is utilized to translate the count-based scores to errors in metric space. For each vertex predicted in contact, its shortest geodesic distance to a ground-truth vertex in contact is calculated. If it is a true positive, this distance is zero. If not, this distance indicates the amount of prediction error along the body.
- **AUC** [29]: The Area under the ROC curve, referred to as AUC, is the most widely used metric for evaluating saliency maps. The saliency map is treated as a binary classifier of fixations at various threshold values (level sets), and a ROC curve is swept out by measuring the true and false positive rates under each binary classifier.
- **aIOU** [40]: IoU is the most commonly used metric for comparing the similarity between two arbitrary shapes. The IoU measure gives the similarity between the predicted region and the ground-truth region, and is defined as the size of the intersection divided by the union of the two regions. It can be formulated as:

$$IoU = \frac{TP}{TP+FP+FN}, \tag{4}$$

  where TP, FP, and FN denote the true positive, false positive, and false negative counts, respectively.
- **SIM** [44]: The similarity metric (SIM) measures the similarity between the prediction map and the ground truth map. Given a prediction map P and a continuous ground truth map Q^D, $SIM(\cdot)$ is computed as the sum of the

minimum values at each element, after normalizing the input maps:

$$SIM(P, Q^D) = \sum_i min(P_i, Q_i^D), \quad where \quad \sum_i P_i = \sum_i Q_i^D = 1. \tag{5}$$

– **MSE** [47]: The Mean Squared Error (MSE) is a measure of the average squared difference between the predicted and actual values in a regression problem. MSE quantifies the average squared difference between the predicted and actual values. It penalizes larger errors more heavily than smaller errors, making it sensitive to outliers. Lower MSE values indicate better model performance in terms of regression accuracy. It is formulated as dividing the total error by n:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} |\hat{y} - y|_2 , \tag{6}$$

where y is the prediction, $\hat{y}$ is the ground truth.

Comparison Methods. Here, we provide an elucidation of the implementation for the methods employed for comparison in the experiment.

– **Baseline**: The baseline model simply takes modality-specific backbones to extract respective features of the image and geometries. Then, it decodes these features separately to obtain the outputs through three branches. This verifies the performance when directly treating this task as a multi-task regression.

– **BSTRO** [19]: BSTRO takes the HRNet as the image backbone and concat the human vertices with the extracted image feature, then it utilizes a multi-layer transformer to estimate the contact vertex. We retain the network architecture while introducing a modification: the vertex of human mesh in BSTRO is downsampled to 431. However, we find that 1723 vertices achieve superior results through experiments. Consequently, during the training, the vertices are downsampled to 1723, the same with LEMON.

– **DECO** [45]: DECO possesses the scene and part context branch to parse the semantics in images, thus facilitating the estimation of human contact. We follow the authors' instructions, taking the Mask2Former [7] to create scene segmentation maps for images in 3DIR. Using the scene and contact branches to train the DECO on 3DIR.

– **3D-AffordanceNet** [11]: 3D-AffordanceNet directly utilizes the DGCNN or PointNet++ to extract per-point features and decodes them to the affordance representation. It tends to anticipate all affordances of the object and may not be consistent with the object affordance in the image. To this end, we slightly modify its structure, taking a cross-attention to update the geometric feature, with the point feature as the query and the image feature as key and value. The affordance representation is obtained by decoding the updated geometric features.

– **IAG-Net** [51]: IAG-Net anticipates the object affordance of objects by establishing the correlations between interaction contents in the image and the geometric features of the object point cloud. We directly utilize the original architecture of IAG-Net to train on the 3DIR. It is worth noting that the training of IAG-Net needs the bounding boxes of the interactive subject and object. We obtain the bounding box by taking the positions of pixels with the smallest and largest in horizontal and vertical coordinates from mask annotations in 3DIR.

– **DJ-RN** [25]: DJ-RN defines the radius for various types of objects, using a sphere to represent the object. Which lifts the spatial relation in images to the 3D space by leveraging the bounding boxes of humans and objects in pixel space and the defined radius. We use its official code to infer the 3DIR data. Additionally, we make minor adjustments to the radius of objects to match the 3D objects in 3DIR (minimal impact on the results).

– **Object pop-up** [39]: This method takes vertices of a posed human as the input, anticipating what objects could interact with the human and the object's spatial position. Since we benchmark it as a comparative method for predicting spatial relation, the object categories may cause ambiguity. Thus, we also take the object point clouds as inputs, integrating the geometric features of humans and objects to make the spatial prediction.

A.3. Training Details

LEMON is implemented by PyTorch and trained with the Adam optimizer. The training epoch is set to 100. All training processes are on 4 NVIDIA 3090 Ti GPUs with an initial learning rate of $1e$-4. The HRNet backbone is initialized with the weights pre-trained on ImageNet [10], while the point cloud extractor is trained from scratch. The hyper-parameters ω_{1-4} that balance loss are set to 40, 40, 20, 20, respectively, and the training batch size is set to 24.

B. Dataset

We curate a question-and-answer (Q & A) list for readers to have a clearer and more detailed understanding of the 3DIR dataset by referring to Datasheets [13].

Q1: For what purpose was the dataset created? Was there a specific gap that needed to be filled? Please provide a description.

A1: 3DIR is collected to facilitate research of 3D human-object interaction relation understanding. It contains paired HOI data and annotations of several interaction elements that elucidate "where" the interaction manifests between the human and object, *e.g.*, human contact, object affordance, and human-object spatial relation. Most existing datasets enable training task-specific models, making the model per-

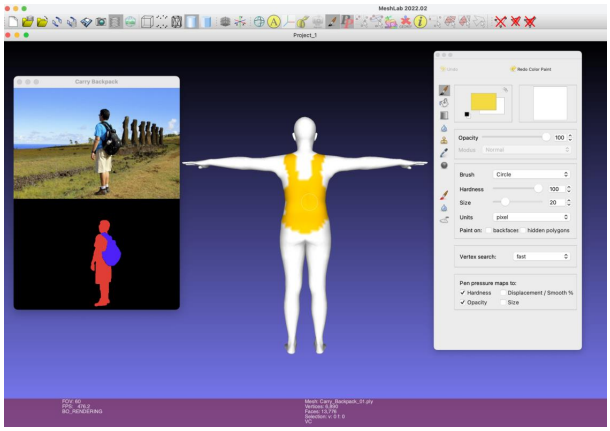

Figure 2. Given the image and mask with an interaction type, drawing vertices in contact with the object.

ceive a certain element of the human-object interaction in isolation. However, interaction elements manifest in the interacting subject, object, and between them. There are also intricate correlations between the interaction elements. We aspire to endow the model with a more comprehensive capacity to understand the interaction relation between humans and objects. Thus, we collect and annotate the 3DIR. Facilitating the perception of human-object interaction relation by enabling the model to learn joint anticipations of interaction elements.

Q2: How many instances are there in total?

A2: There are 5001 in-the-wild images with explicit interaction content, and the quantity of 3D object instances in 3DIR is 5143. The whole data spans 21 object classes and 17 interaction categories. Besides, there are over $25k$ multiple annotations for these collected data.

Q3: What data does each instance consist of?

A3: A sample of 3DIR contains the following data: 1) an interaction image; 2) an object point cloud; 3) the SMPL-H pseudo-GTs; 4) masks of the interacting human and object in the image; 5) 3D affordance annotation of the object; 6) dense human contact label; 7) spatial position of object centers in the same camera coordinate of the fitted human. In addition, taking the masks could easily calculate the bounding boxes of the human and object.

Q4: Why chose the SMPL-H as the human mesh?

A4: Firstly, SMPL-H and SMPL [30] share the same topology, so the model trained with SMPL-H could directly generalize to SMPL. However, SMPL lacks parameters for the hands, resulting in some types of grasping interactions that cannot be presented. Another human model SMPL-X [38], also includes hand parameters. But it has 10475 vertices and nearly 50% of the vertices are on the head, which usually does not greatly impact the interaction. To save the computational overhead, we ultimately chose to use the OSX [27] to fit the SMPL-H human mesh.

Q5: Why use the OSX to fit the human body, and how to implement the pipeline?

A5: In natural interaction images, sometimes only the upper body of a human is visible, and the model should possess the ability to predict the human mesh for these cases. OSX [27] is trained on multiple datasets, such as MPII [1], Human3.6M [21], AGORA [37]. It masters the prior knowledge of human topology with the training on the above datasets. In addition, it provides a pipeline to fit the UBody dataset. The models trained on UBody perform well in predicting situations where only the upper body is present. Therefore, we use the same pipeline as UBody to fit the human mesh of 3DIR. To maximize the utilization of the pre-trained OSX, we still retained the face decoder, and the SMPL-X humans are transferred to SMPL-H through the official script in SMPL-X [38].

Q6: What mechanisms or procedures were used to collect and annotate the data (*e.g.*, software program, software API)?

A6: Images are collected from HAKE [26], V-COCO [15], PIAD [51] and websites with free licenses. The 3D object instances mainly come from PartNet [32], 3D-AffordanceNet [11] and Objaverse [9]. The objects selected from Objaverse are downloaded through its official API. For multiple annotations, we used several tools to implement them respectively. 1) We leverage the **ISAT** [52] to annotate the human and object masks. It integrates the SAM [24] and can perform interactive semi-automatic annotation. 2) The software **MeshLab** [36] is utilized to "draw" the dense human contact vertices. In which mesh could be translated, rotated, and zoomed in/out, it also supports drawing colors on the mesh vertices. As shown in Fig. 2, we pin the image and mask onto the screen, and annotators color the vertices on the human body that are in contact with the object. The contact vertices are captured based on their color through a script. 3) We refer to the 3D-AffordanceNet [11] to annotate the object affordance. The object instances are imported to the MeshLab and we color the affordance key points and the propagable region. Their coordinates are recorded, and the remaining propagation steps and algorithms are consistent with 3D-AffordanceNet. Please refer to it for more details. 4) With the contact annotation, we color the fitted human mesh and import it into the **Blender** [8] to annotate the object's spatial position. Specifically, we create geometric and material nodes for annotators to view the contact on the human body. In conjunction with the image, annotators need to adjust the position of the sphere (object proxy) with predefined radii to align the human-object spatial relation in the image, as shown in Fig. 3. The criteria for defining the radius of the object is clarified in **Q7**.

Q7: How to define the radius of the object's proxy sphere? Please provide a detailed description.

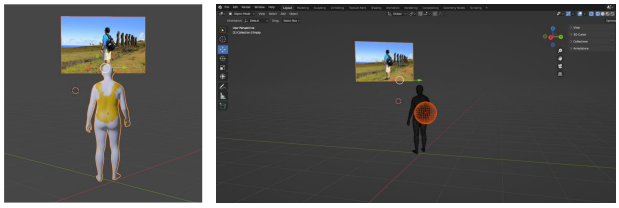

Figure 3. Adjust the spatial position of the proxy sphere to align with human-object spatial relation in the image by referring to the image and contact vertices.

Table 2. The defined radii for each object category, unit: m.

Backpack	0.265	**Bottle**	0.140	**Mug**	0.094
Baseballbat	0.325	**Suitcase**	0.332	**Vase**	0.197
Skateboard	0.375	**Bicycle**	0.675	**Bowl**	0.132
Tennisracket	0.298	**Scissors**	0.179	**Chair**	0.455
Surfboard	0.687	**Keyboard**	0.217	**Knife**	0.173
Motorcycle	0.710	**Earphone**	0.132	**Bag**	0.192
Umbrella	0.372	**Guitar**	0.394	**Bed**	1.154

A7: We define the object's proxy radius relative to the fitted human mesh. As the fitted human body's unit measurements in the coordinate system match the real world, defining objects' radii in this way closely aligns with the actual size of objects. For each object category, we import 20 instances into Blender and scale them to match the human size. Then, the center of the proxy sphere is moved to the object's geometric center, adjusting the sphere radius to just envelop the object. We record each radius and calculate the mean as the proxy sphere radius for each object category. Note that these radii could be used as a basis for further tuning through the mask ratio of humans and objects, the ratio of bounding boxes in the image, or others. The defined radii are shown in Tab. 2.

Q8: When fitting the human body, there are already assumed camera optical centers and focal lengths. Why not directly project the center of the object in the 2D image into 3D space?

A8: The spatial annotation of the object center is conducted within the same camera coordinates as fitted humans. For in-the-wild images, HMR (human mesh recovery) methods mostly take weak-perspective camera models to project the human mesh [18, 22] to the image plane. Which considers the depth to be relatively uniform for the human instance. Therefore, directly back-projecting the objects' centers from images into 3D space will cause depth ambiguity. This is also explained in DJ-RN [25], where they further determine the depth of objects by defining their radii. For HOIs, the relative depth between humans and objects is crucial for representing their spatial relation. So, based on the camera scale s and translation t inferred by OSX, we manually annotate the objects' center positions at the same camera coordinates as the fitted humans.

Q9: How to ensure the quality of annotation in the processes?

A9: We conduct subjective cross-checks and objective measuring to ensure the quality of the annotations. In specific, we initially release the annotation requirements and recruit a cohort of annotators. For the contact annotation, we provide 100 author-annotated instances, as well as detailed annotation instructions and software tutorials. Then, the annotators make annotations for these 100 samples following the instructions. Referring to DAMON [45], the Intersection-over-Union (IOU) is calculated between the author-annotation and annotations made by candidate annotators. For the objects' spatial positions, we select 100 instances from the BEHAVE [3] dataset and record the coordinates of the objects' geometric centers. The annotators are required to annotate the objects' positions according to the manner in **Q6**. The MSE is calculated between the ground truth in BEHAVE and annotations. Eventually, we select 5 qualified annotators through the evaluation results. These 5 annotators conduct three rounds of annotation, with each round involving subjective cross-check and author-check to filter out instances with glaring annotation errors. Besides, for each instance, we cyclically use the annotation of each annotator as a temporary reference and calculate the measurement metrics (IOU, MSE) between it and the remaining annotations. A re-annotation process will be initiated if there is significant variance among the metrics. We choose the temporary annotation with the minimum variance in metrics as the final annotation. The object affordances are annotated by authors. We train IAG-Net [51] on 10 object categories we annotated and 11 categories selected from 3D-AffordanceNet. The AUC and aIOU for our annotated data are $85.15, 37.93$, while for the selected data are $85.76, 37.82$. This indirectly indicates that the quality of our annotations is comparable to existing annotations, which could effectively support the model training.

Q10: Are there any errors, noise, or redundancies in the dataset? If so, please provide a description.

A10: Since some annotations are based on human knowledge, *e.g.*, human contact, some annotations may not be completely accurate. However, these will not seriously impact the final results. Similar to those instances in the DAMON [45] and HOT [6] dataset.

Q11: Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or includes the content of individuals non-public communications)?

A11: No.

Q12: Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety?

A12: No.

Table 3. Comparison of contact estimation on DAMON dataset.

Method	Precision	Recall	F1	geo. (cm)
BSTRO	0.57	0.58	0.53	28.81
DECO	0.64	0.60	0.59	19.98
Ours P.	0.69	0.62	0.61	14.75
Ours D.	0.67	0.66	0.63	13.45

Table 4. Take BEHAVE as the unseen dataset. **(a)** Comparison of contact estimation on BEHAVE dataset. **(b)** Comparison of spatial prediction on BEHAVE dataset.

Method	Precision	Recall	F1	geo. (cm)
BSTRO	0.03	0.11	0.05	48.37
DECO	0.13	0.25	0.17	40.45
Ours P.	0.17	0.27	0.20	35.14
Ours D.	0.19	0.30	0.21	29.82

(a)

Method	MSE
DJ-RN	0.287
PopUp	0.149
Ours P.	0.098
Ours D.	0.084

(b)

Table 5. Object affordance anticipation on PIAD dataset. Seen and Unseen are two settings, 3D-Aff. denotes 3D-AffordanceNet.

Method	Seen			Unseen		
	AUC	aIOU	SIM	AUC	aIOU	SIM
3D-Aff.	83.07	16.65	0.46	59.51	3.87	0.323
IAG	84.85	20.51	0.54	64.14	7.35	0.346
Ours P.	85.64	22.98	0.56	65.96	7.88	0.351
Ours D.	86.07	23.16	0.56	66.22	8.23	0.355

C. Experiments

We conduct more experiments to verify the superiority of the LEMON. The details are described as follows.

C.1. Test on Multiple Datasets

DAMON Dataset [45]. We select around $3k$ data from DAMON that match the object categories in 3DIR, divide the training and testing sets in an $8:2$ ratio and train the comparison methods. The training of DECO here is consistent with its original architecture, using the scene, part, and contact branches. The human meshes used in LEMON are directly inferred through OSX accompanied by a transferring process, without the fitting pipeline that is illustrated in Sec. B **Q5**. Evaluation results are recorded in Tab. 3.

BEHAVE Dataset [3]. The BEHAVE dataset is utilized as an unseen dataset to evaluate both the contact estimation and spatial prediction. For the object position, we record the geometry center of aligned objects in the BEHAVE. Each comparison method is only trained on the 3DIR dataset and tested on the BEHAVE dataset. The metrics of both the contact and spatial prediction are shown in Tab. 4.

PIAD Dataset [51]. We conduct tests with two settings on PIAD. 1) Seen: as many images in the PIAD dataset do not include complete humans, the template human is utilized to train LEMON on PIAD. 2) Unseen: PIAD and 3DIR have 11 overlapping object categories, so we train methods on 10 categories in 3DIR (not included in PIAD) and test them on the 11 categories (regarded as unseen data) in PIAD. The results of "Seen" and "Unseen" settings are recorded in Tab. 5.

Following DECO [45] and HOT [6], we evaluate whether the estimated contact by LMEON benefits human pose and shape (HPS) regression. The test is on the PROX "quantitative" dataset [16], and the experimental setup is the same with DECO and HOT. Since LEMON focuses on estimating vertices that are in contact with objects, the neglect of estimating the contact between feet and the ground has a certain impact on the results. Thus, we fine-tune LEMON on the DAMON dataset and give the results in Tab. 6.

Table 6. HPS estimation performance using contact derived from different sources.

Methods	✗ Contact	Prox	HOT	DECO	LEMON	GT
V2V ↓	183.3	174.0	172.3	171.6	**170.9**	163.0

Table 7. The influence of learning rate (Lr) and batch size (B). The best results are covered with the mask.

Lr	B	Contact				Affordance			Spatial
		Pre.	Rec.	F1	geo.	AUC	aIOU	SIM	MSE
1e-3	16	0.76	0.77	0.75	8.67	87.26	40.52	0.59	0.017
1e-4	16	0.76	0.80	0.76	7.98	87.97	40.86	0.62	0.013
1e-5	16	0.78	0.79	0.76	7.85	88.02	41.10	0.62	0.014
1e-3	24	0.77	0.77	0.78	7.86	87.69	40.77	0.62	0.013
1e-4	24	0.78	0.82	0.78	7.55	88.51	41.34	0.64	0.010
1e-5	24	0.76	0.75	0.75	8.14	87.57	40.63	0.59	0.016
1e-3	32	0.79	0.80	0.78	7.63	88.35	41.28	0.64	0.010
1e-4	32	0.78	0.80	0.77	7.67	88.29	41,02	0.62	0.012

Table 8. Performance of the model under different quantities of human vertices.

Vertices	Contact				Affordance			Spatial
	Pre.	Rec.	F1	geo.	AUC	aIOU	SIM	MSE
431	0.75	0.78	0.74	12.13	87.52	39.89	0.61	0.012
1723	0.78	0.82	0.78	7.55	88.51	41.34	0.64	0.010
6890	0.79	0.80	0.78	7.62	88.32	41.53	0.63	0.009

Table 9. Performance of object spatial position prediction when using different positional encoding methods. P.E. indicates positional encoding.

	w/o P.E.	Learnable	Sine & Cosine	Relative
MSE	0.017	0.010	0.012	0.014

Table 10. Evaluation metrics for each object. Ear. is Earphone, Baseb. is Baseballbat, Tenn. is Tennisracket, Motor. is Motorcycle, Back. is Backpack, Kni.is Knife, Bicy. is Bicycle, Umbr. is Umbrella, Keyb. is Keyboard, Bott. is Bottle, Surf. is Surfboard, Suitc. is Suitcase, Skate. is Skateboard. P. indicates take PointNet++ as the point cloud backbone while D. indicates DGCNN.

	Metr.	Ear.	Baseb.	Tenn.	Bag	Motor.	Gui.	Back.	Chair	Kni.	Bicy.	Umbr.	Keyb.	Scis.	Bott.	Bowl	Surf.	Mug	Suitc.	Vase	Skate.	Bed
LEMON P.	**Prec.**	0.97	0.75	0.80	0.29	0.90	0.73	0.72	0.77	0.75	0.84	0.70	0.67	0.35	0.71	0.76	0.85	0.67	0.93	0.73	0.79	0.75
	Rec.	0.94	0.87	0.86	0.43	0.88	0.75	0.69	0.89	0.84	0.86	0.86	0.84	0.54	0.83	0.81	0.84	0.59	0.94	0.84	0.94	0.77
	F1	0.95	0.79	0.8	0.33	0.88	0.72	0.69	0.79	0.78	0.84	0.76	0.73	0.34	0.73	0.75	0.84	0.60	0.94	0.76	0.84	0.74
	geo.	1.54	2.94	13.75	37.27	2.34	3.30	7.25	3.97	16.39	6.26	22.08	18.65	32.75	24.19	3.37	5.70	19.77	0.25	3.39	3.91	4.38
	AUC	86.85	94.41	97.09	92.08	97.67	96.28	90.49	94.32	82.67	83.78	95.78	86.2	64.86	68.44	68.55	77.43	82.75	91.4	69.77	91.98	89.71
	aIOU	23.71	63.12	58.12	38.76	51.64	71.05	50.8	36.18	9.62	35.50	59.24	15.06	5.12	11.99	3.63	42.05	32.029	50.42	5.02	76.21	23.40
	SIM	0.61	0.74	0.70	0.47	0.63	0.82	0.64	0.71	0.60	0.44	0.66	0.26	0.41	0.60	0.79	0.39	0.64	0.55	0.67	0.85	0.59
	MSE	0.016	0.012	0.028	0.022	0.012	0.009	0.032	0.01	0.008	0.007	0.010	0.008	0.008	0.006	0.004	0.012	0.008	0.018	0.003	0.008	0.048
LEMON D.	**Prec.**	0.95	0.77	0.83	0.38	0.89	0.74	0.77	0.81	0.80	0.82	0.74	0.69	0.37	0.75	0.80	0.85	0.68	0.95	0.74	0.80	0.78
	Rec.	0.95	0.87	0.84	0.46	0.89	0.73	0.69	0.86	0.79	0.89	0.87	0.79	0.67	0.78	0.81	0.82	0.76	0.94	0.82	0.89	0.79
	F1	0.95	0.80	0.82	0.40	0.88	0.71	0.69	0.80	0.78	0.84	0.78	0.71	0.42	0.74	0.78	0.82	0.69	0.94	0.76	0.83	0.77
	geo.	0.70	2.47	12.18	31.53	2.78	3.51	4.02	3.11	9.94	6.62	23.97	16.09	27.43	17.91	3.59	5.86	11.87	0.04	2.92	1.95	2.73
	AUC	87.90	97.45	98.99	93.04	97.92	97.74	95.49	94.23	82.02	87.10	96.88	86.67	75.96	69.16	66.69	73.67	83.16	84.83	72.08	93.64	87.66
	aIOU	22.01	69.37	71.03	43.84	53.04	72.65	60.33	36.68	9.56	31.09	60.59	16.72	5.46	12.76	3.48	41.36	32.94	41.45	5.29	79.70	21.63
	SIM	0.61	0.81	0.78	0.48	0.64	0.84	0.7	0.69	0.61	0.40	0.69	0.28	0.46	0.60	0.79	0.37	0.65	0.51	0.68	0.87	0.57
	MSE	0.002	0.007	0.025	0.019	0.008	0.007	0.021	0.011	0.005	0.007	0.012	0.006	0.004	0.006	0.003	0.008	0.003	0.030	0.002	0.005	0.040

C.2. Hyperparameters

During the training process, some hyperparameters have impacts on the model performance. We provide a series of experiments to determine the ultimate hyperparameters. All experiments are conducted when taking DGCNN as the point cloud backbone. Concerning the impact of learning rate and batch size on the model, we conduct comparative experiments by adjusting the learning rate across orders of magnitude and combining it with different batch sizes. The results are presented in Tab. 7. Besides, we test whether the quantity of human vertices influences the model performance. The quantity of object points is consistent with 3D-AffordanceNet and IAG-Net. The results are shown in Tab. 8. As can be seen, when the number of vertices increases from 431 to 1723, there is a significant increase in model performance. However, the growth is not significant when increasing from 1723 to 6890. To conserve computational overhead, we ultimately chose to sample the number of human vertices at 1723. We also test the performance of several position encoding methods in object spatial position prediction. The results are reported in Tab. 9. For w_{1-4} that balance the loss, we test them according to the order of magnitude and multiples, and finally determine their specific values. Due to the excessive number of combinations, we do not exhibit the results one by one here.

C.3. More Results

In the main paper, we provide overall results of metrics and some visualization results of LEMON on the 3DIR benchmark. Here, we show evaluation metrics for each object category and more visualization results that are not presented in the main paper. The metrics for each object are shown in Tab. 10. Fig. 4 and Fig. 5 demonstrate more visual results, including human contact, object affordance, and spatial relation. For the experiments in Tab. 4, we also provide some visual results of LEMON, shown in Fig. 6.

D. Application Prospect

The human contact, object affordance, and human-object spatial relation are crucial elements for representing the human-object interaction relation. Perceiving these elements also links the HOI understanding with downstream applications.

Embodied AI [43]. One characteristic of embodied intelligence is to learn and improve skills by actively interacting with the surrounding environment [33]. However, the prerequisite for actively interacting with the environment is the ability to perceive or understand how to interact [12, 34]. Learning and understanding interactions from human-object interaction is an effective manner. The interaction elements reflect how the interaction is manifest at the counterparts. For example, object affordance represents what action could be done for the object and which location supports the action, revealing "where to interact". Human contact represents the regions capable of interacting with objects on the embodiment, revealing "where are utilized to interact". Spatial relation connects the interacting subject and object. These elements collectively formulate the interaction relation. Perceiving interaction elements enables the embodied agent to make policies on how to interact with the environment, thereby learning from interactions.

Interaction Modeling. Modeling or recovering the in-

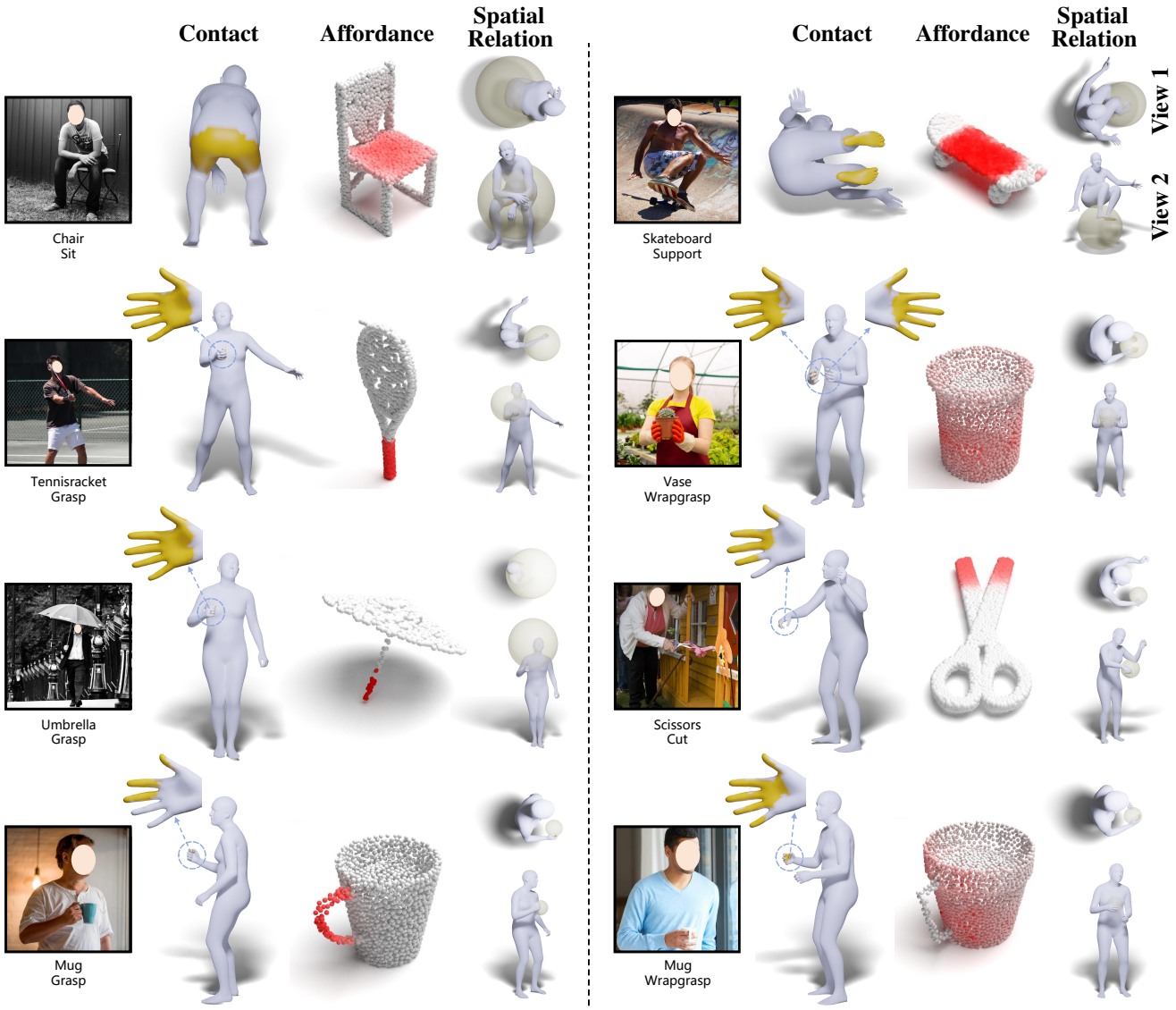

Figure 4. More visual results anticipated by LEMON, including human contact, object affordance, and human-object spatial relation.

teraction is a significant but extremely challenging task. It holds significant prospects for application in the animation and digital avatar industries. Some methods have explored the benefits of human contact in interaction modeling [17, 48–50, 53]. But obviously, human contact is only one aspect of the interaction relation. In addition to this, object affordance and human-object spatial relation also offer clues for interaction modeling. The exploration of incorporating more comprehensive representations of interaction relation (*e.g.*, affordance, spatial relation) into interaction modeling is a worthy investigated future research direction. Which may further advance the modeling of interactions.

Augement & Virtual Reality. With the emergence of immersive spatial computing devices, *e.g.*, Meta Quest, Apple Vision Pro [46], and PICO, AR/VR will permeate many industries such as education, healthcare, gaming, and so on. The way of human-computer interaction (HCI) [5, 31] is a crucial symptom node for these devices. Perceiving 3D human-object interaction relation in the virtual or augmented world provides feedback signals to adjust the manner of HCI, thereby enhancing the user's immersion.

Imitation Learning [20]. Imitation learning is an important way to drive robots to complete certain tasks, which makes intelligent agents perform interactions by observing demonstrations from humans or other sources. The interaction elements perceived from 3D human-object interactions offer explicit representations to reveal "what" interaction could be performed with an object and "how" to interact with it. These rich interaction priors enhance the machine's ability to imitate the interaction manners, thereby learning skills from them. Which is beneficial for configurations like dexterous hand [2, 28] and humanoid robot [4, 23, 35].

Figure 5. More visual results anticipated by LEMON, including human contact, object affordance, and human-object spatial relation.

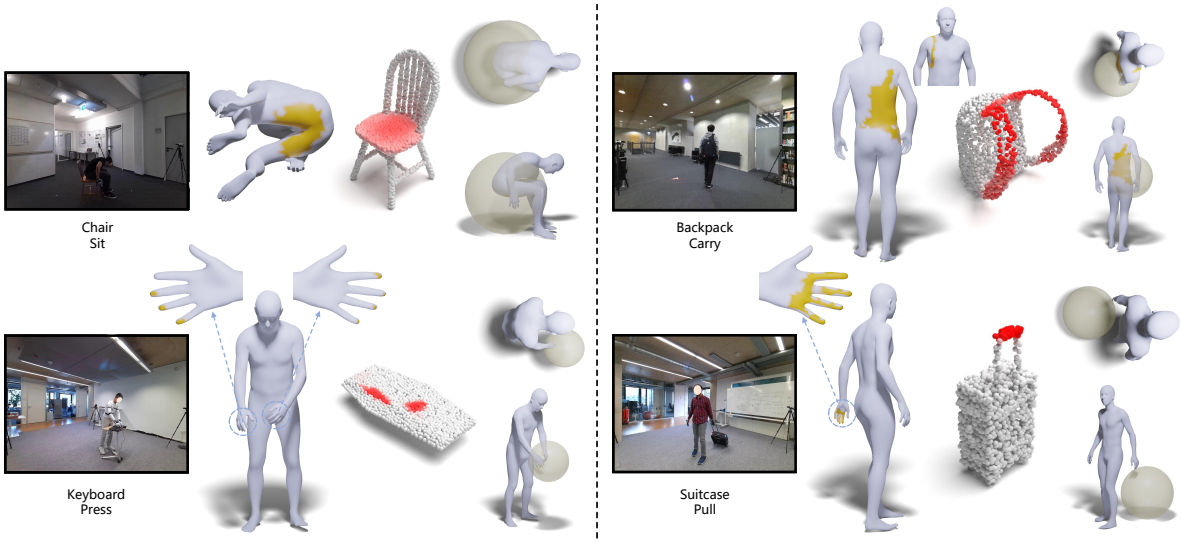

Figure 6. Anticipating on the unseen BEHAVE data, including human contact, object affordance, and human-object spatial relation.

References

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 4

[2] OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020. 8

[3] Bharat Lal Bhatnagar, Xianghui Xie, Ilya Petrov, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Behave: Dataset and method for tracking human object interactions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2022. 5, 6

[4] Cynthia Breazeal. Emotion and sociable humanoid robots. *International journal of human-computer studies*, 59(1-2): 119–155, 2003. 8

[5] John M Carroll. Human-computer interaction: psychology as a science of design. *Annual review of psychology*, 48(1): 61–83, 1997. 8

[6] Yixin Chen, Sai Kumar Dwivedi, Michael J. Black, and Dimitrios Tzionas. Detecting human-object contact in images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 5, 6

[7] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. 2022. 3

[8] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. 4

[9] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 4

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 3

[11] Shengheng Deng, Xun Xu, Chaozheng Wu, Ke Chen, and Kui Jia. 3d affordancenet: A benchmark for visual object affordance understanding. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1778–1787, 2021. 1, 2, 3, 4

[12] Kuan Fang, Te-Lin Wu, Daniel Yang, Silvio Savarese, and Joseph J Lim. Demo2vec: Reasoning object affordances from online videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2139–2147, 2018. 7

[13] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92, 2021. 3

[14] Cyril Goutte and Eric Gaussier. A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In *European conference on information retrieval*, pages 345–359. Springer, 2005. 2

[15] Saurabh Gupta and Jitendra Malik. Visual semantic role labeling. *arXiv preprint arXiv:1505.04474*, 2015. 4

[16] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J Black. Resolving 3d human pose ambiguities with 3d scene constraints. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2282–2292, 2019. 6

[17] Mohamed Hassan, Partha Ghosh, Joachim Tesch, Dimitrios Tzionas, and Michael J Black. Populating 3d scenes by learning human-scene interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14708–14718, 2021. 8

[18] Radu Horaud, Fadi Dornaika, and Bart Lamiroy. Object pose: The link between weak perspective, paraperspective, and full perspective. *International Journal of Computer Vision*, 22(2):173–189, 1997. 5

[19] Chun-Hao P. Huang, Hongwei Yi, Markus Höschle, Matvey Safroshkin, Tsvetelina Alexiadis, Senya Polikovsky, Daniel Scharstein, and Michael J. Black. Capturing and inferring dense full-body human-scene contact. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 13274–13285, 2022. 1, 2, 3

[20] Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne. Imitation learning: A survey of learning methods. *ACM Computing Surveys (CSUR)*, 50(2):1–35, 2017. 8

[21] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013. 4

[22] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7122–7131, 2018. 5

[23] Kenji Kaneko, Kensuke Harada, Fumio Kanehiro, Go Miyamori, and Kazuhiko Akachi. Humanoid robot hrp-3. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2471–2478. IEEE, 2008. 8

[24] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 4

[25] Yong-Lu Li, Xinpeng Liu, Han Lu, Shiyi Wang, Junqi Liu, Jiefeng Li, and Cewu Lu. Detailed 2d-3d joint representation for human-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10166–10175, 2020. 3, 5

[26] Yong-Lu Li, Liang Xu, Xinpeng Liu, Xijie Huang, Yue Xu, Shiyi Wang, Hao-Shu Fang, Ze Ma, Mingyang Chen, and Cewu Lu. Pastanet: Toward human activity knowledge engine. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 382–391, 2020. 4

[27] Jing Lin, Ailing Zeng, Haoqian Wang, Lei Zhang, and Yu Li. One-stage 3d whole-body mesh recovery with component aware transformer. *CVPR*, 2023. 4

[28] Hong Liu, Ke Wu, Peter Meusel, Nikolaus Seitz, Gerd Hirzinger, MH Jin, YW Liu, SW Fan, T Lan, and ZP Chen. Multisensory five-finger dexterous hand: The dlr/hit hand ii. In *2008 IEEE/RSJ international conference on intelligent robots and systems*, pages 3692–3697. IEEE, 2008. 8

[29] Jorge M Lobo, Alberto Jiménez-Valverde, and Raimundo Real. Auc: a misleading measure of the performance of predictive distribution models. *Global ecology and Biogeography*, 17(2):145–151, 2008. 2

[30] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 851–866. 2023. 4

[31] I Scott MacKenzie. Human-computer interaction: An empirical research perspective. 2012. 8

[32] 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. 4

[33] Tushar Nagarajan and Kristen Grauman. Learning affordance landscapes for interaction exploration in 3d environments. *Advances in Neural Information Processing Systems*, 33:2005–2015, 2020. 7

[34] Tushar Nagarajan, Christoph Feichtenhofer, and Kristen Grauman. Grounded human-object interaction hotspots from video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8688–8697, 2019. 7

[35] Gabe Nelson, Aaron Saunders, and Robert Playter. The petman and atlas robots at boston dynamics. *Humanoid robotics: A reference*, pages 169–186, 2018. 8

[36] Cignoni Paolo, Muntoni Alessandro, Ranzuglia Guido, and Callieri Marco. Meshlab. 4

[37] Priyanka Patel, Chun-Hao P Huang, Joachim Tesch, David T Hoffmann, Shashank Tripathi, and Michael J Black. Agora: Avatars in geography optimized for regression analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13468–13478, 2021. 4

[38] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 10975–10985, 2019. 4

[39] Ilya A Petrov, Riccardo Marin, Julian Chibane, and Gerard Pons-Moll. Object pop-up: Can we infer 3d objects and their poses from human interactions alone? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 2, 3

[40] Md Atiqur Rahman and Yang Wang. Optimizing intersection-over-union in deep neural networks for image segmentation. In *International symposium on visual computing*, pages 234–244. Springer, 2016. 2

[41] Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J. Black. Generating 3D faces using convolutional mesh autoencoders. In *European Conference on Computer Vision (ECCV)*, pages 725–741, 2018. 1

[42] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), 2017. 1

[43] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347, 2019. 7

[44] Michael J Swain and Dana H Ballard. Color indexing. *International journal of computer vision*, 7(1):11–32, 1991. 2

[45] Shashank Tripathi, Agniv Chatterjee, Jean-Claude Passy, Hongwei Yi, Dimitrios Tzionas, and Michael J. Black. DECO: Dense estimation of 3D human-scene contact in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8001–8013, 2023. 2, 3, 5, 6

[46] Ethan Waisberg, Joshua Ong, Mouayad Masalkhi, Nasif Zaman, Prithul Sarker, Andrew G Lee, and Alireza Tavakkoli. The future of ophthalmology and vision science with the apple vision pro. *Eye*, pages 1–2, 2023. 8

[47] Zhou Wang and Alan C Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE signal processing magazine*, 26(1):98–117, 2009. 3

[48] Xianghui Xie, Bharat Lal Bhatnagar, and Gerard Pons-Moll. Chore: Contact, human and object reconstruction from a single rgb image. In *European Conference on Computer Vision (ECCV)*. Springer, 2022. 8

[49] Sirui Xu, Zhengyuan Li, Yu-Xiong Wang, and Liang-Yan Gui. Interdiff: Generating 3d human-object interactions with physics-informed diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14928–14940, 2023.

[50] Xiang Xu, Hanbyul Joo, Greg Mori, and Manolis Savva. D3d-hoi: Dynamic 3d human-object interactions from videos. *arXiv preprint arXiv:2108.08420*, 2021. 8

[51] Yuhang Yang, Wei Zhai, Hongchen Luo, Yang Cao, Jiebo Luo, and Zheng-Jun Zha. Grounding 3d object affordance from 2d interactions in images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10905–10915, 2023. 1, 2, 3, 4, 5, 6

[52] Alias-z yatengLG and horffmanwang. Isat with segment anything: Image segmentation annotation tool with segment anything, 2023. Open source software available from $https://github.com/yatengLG/ISAT_with_segment_anything$. 4

[53] Jason Y Zhang, Sam Pepose, Hanbyul Joo, Deva Ramanan, Jitendra Malik, and Angjoo Kanazawa. Perceiving 3d human-object spatial arrangements from a single image in the wild. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, pages 34–51. Springer, 2020. 8