H2O: Two Hands Manipulating Objects for First Person Interaction Recognition

Taein Kwon[1], Bugra Tekin[2], Jan Stühmer[*3], Federica Bogo[2], and Marc Pollefeys[1,2]

[1]ETH Zürich, [2]Microsoft, [3]Samsung AI Center, Cambridge

2021 IEEE/CVF International Conference on Computer Vision (ICCV) | 978-1-6654-2812-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICCV48922.2021.00998

Abstract

We present a comprehensive framework for egocentric interaction recognition using markerless 3D annotations of two hands manipulating objects. To this end, we propose a method to create a unified dataset for egocentric 3D interaction recognition. Our method produces annotations of the 3D pose of two hands and the 6D pose of the manipulated objects, along with their interaction labels for each frame. Our dataset, called H2O (2 Hands and Objects), provides synchronized multi-view RGB-D images, interaction labels, object classes, ground-truth 3D poses for left & right hands, 6D object poses, ground-truth camera poses, object meshes and scene point clouds. To the best of our knowledge, this is the first benchmark that enables the study of first-person actions with the use of the pose of both left and right hands manipulating objects and presents an unprecedented level of detail for egocentric 3D interaction recognition. We further propose the method to predict interaction classes by estimating the 3D pose of two hands and the 6D pose of the manipulated objects, jointly from RGB images. Our method models both inter- and intra-dependencies between both hands and objects by learning the topology of a graph convolutional network that predicts interactions. We show that our method facilitated by this dataset establishes a strong baseline for joint hand-object pose estimation and achieves state-of-the-art accuracy for first person interaction recognition.

1. Introduction

In recent years, there has been tremendous progress in video understanding and action recognition. Current algorithms can reliably recognize the action the subject is performing in many unconstrained settings from third person viewpoints [9, 22, 23, 24, 71, 89]. Although action recognition from first-person views has many applications in augmented reality, robotics and surveillance, it trails behind the progress in third person views, mostly due to the lack of large and diverse egocentric datasets. From an egocentric viewpoint, action recognition is mostly about understanding hand & object interactions. A unified understanding of the positions and movements of hands and the manipulated objects is crucial for recognizing egocentric interactions. However, existing first-person interaction datasets mostly provide only 2D features (*e.g.* bounding boxes, hand segmentation) without reasoning in 3D about the motions of hands and the manipulated objects. In this work, we propose, for the first time, a unified dataset for first person interaction recognition with markerless 3D annotations of two hands manipulating objects, as depicted in Fig. 1. We collect a richly annotated dataset including synchronized RGB-D images, camera poses, right & left hand poses, object poses, object meshes, scene point clouds and action labels, which provides an unprecedented level of detail for understanding 3D hand-object interactions. With the help of our dataset, we present the first method to estimate jointly the 3D pose of two hands and objects from a color image. We further propose to learn interdependencies within and across hand and object poses using an adaptive graph convolutional network for 3D interaction recognition.

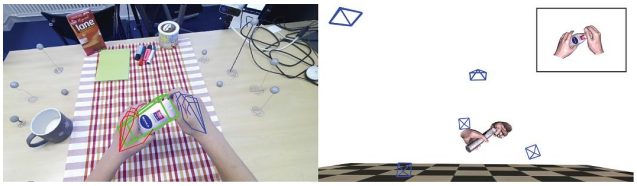


Figure 1: **Two hands manipulating objects for first person interaction recognition.** We propose a dataset providing rich annotations for 3D poses of left & right hands, 6D object poses, camera poses, object meshes and scene point clouds, along with their associated interaction labels. We leverage our dataset to propose novel methods for 3D interaction recognition.

Jointly capturing hands in action and the manipulated objects in 3D is a challenging problem due to reciprocal occlusions. The problem is more challenging from first person viewpoints due to the unique challenges brought by egocentric vision such as fast camera motion, large occlusion, background clutter [49] and most importantly, lack of datasets. Recent works have proposed datasets that successfully addressed some of these challenges. Sridhar et al. [73] have presented one of the earliest datasets for hand-object interactions, in which a single hand manipulates a cuboid object. Pioneering works by [26, 32, 34] have further proposed datasets that include 3D annotations for object ma-

[*]Work performed while at Microsoft.
Project Page: https://www.taeinkwon.com/projects/h2o

978-1-6654-2812-5/21/$31.00 ©2021 IEEE
DOI 10.1109/ICCV48922.2021.00998

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 14:25:18 UTC from IEEE Xplore. Restrictions apply.

nipulation scenarios of a single hand.

Most of these works, however, are limited by different factors. They mainly focus on *single hand manipulation scenarios* [26, 32, 34]. While single hand manipulation is relevant for some scenarios, most of the time, hand-object interaction involves two hands manipulating an object. Using *only 2D annotations*, [79, 80] presented datasets for hand-hand and hand-object interactions. The intricate nature of hand-object interactions, however, requires 3D reasoning rather than 2D to better resolve mutual occlusions. In the context of hand-object interactions, early work mostly tackles the problem of joint estimation of 3D hand and object poses, *without reasoning about the actions*. While precise 3D position data for hands and objects is crucial for many applications in robotics and graphics, the sole knowledge of the pose lacks semantic meaning about the actions of the subject. To that end, [26] released an egocentric action dataset including 3D annotations of hands and objects; however, the data is captured with an *intrusive motion capture system*. Although motion capture datasets [26, 75] can provide large amounts of training samples with accurate 3D annotations, they can only be captured in controlled settings and have visible markers on the images that bias pose prediction in color images. *Synthetic datasets* [34] could provide an alternative to them, however, the existing ones cannot yet reach the realism that is needed to generalize to real images and are only for single-image scenarios that lack temporal context crucial for recognizing interactions.

Our method aims at tackling these limitations exhibited by prior work. To this end, we propose an approach for creating a unified dataset for egocentric 3D interaction recognition that includes markerless annotations of the 3D pose of two hands and the 6D pose of the manipulated objects, along with their associated action labels for each frame of a large number of recordings that include 571,645 synchronized RGB-D frames. In addition, we propose the first method to jointly predict the 3D pose of two hands and 6D pose of the manipulated objects using only RGB images and present a novel 3D interaction recognition approach that learns the interdependencies between hand and object poses by a topology-aware graph convolutional network.

Our contributions can be listed as follows:

- We present the first unified dataset for egocentric interaction recognition with markerless 3D annotations of two hands and the 6D pose of manipulated objects. Our dataset, which we call $H2O$, standing for *2 hands* and *objects*, provides rich ground-truth annotations for 3D hand-object poses & shapes, action labels, camera poses, scene point clouds and object meshes that enable us to produce comprehensive egocentric scene interpretations.
- We propose a semi-automatic pipeline to curate a hand-object interaction dataset with action labels and the poses of two interacting hands as well as the objects in contact, using a practical multi-camera system with diverse backgrounds. We demonstrate the fidelity and accuracy of our annotations by detailed verifications.
- We introduce a unified approach to recognize hand-object interactions from RGB images that simultaneously predicts, for the first time, the 3D pose of two interacting hands and the 6D pose of manipulated objects, along with action and object classes.
- Leveraging our dataset, we propose a novel method for 3D interaction recognition that learns the interdependencies between two hands and objects with a topology-aware graph convolutional network. To this end, we parameterize both hand and object poses as individual graphs and combine them in a single multi-graph architecture. We then learn the interdependencies and connections between different graph entities with an adaptive architecture and compute the topology of the multi-graph structure for recognizing 3D hand-object interactions.

We demonstrate that using the pose predictions facilitated by our dataset, we achieve better overall performance for recognizing interactions outperforming the state-of-the-art [9, 17, 23]. We further provide baselines for hand & object pose estimation and interaction recognition to enable further benchmarking on this dataset. We will make our dataset and annotations publicly available upon acceptance.

2. Related Work

Datasets for egocentric action recognition and hand-object pose estimation. While many datasets for third-person action recognition have been proposed throughout the years [30, 40, 69, 96], recently there is a surge in interest for data targeting also egocentric scenarios [12, 13, 29, 47, 60, 68] that involve mostly 2D features. These datasets provide only limited multi-view data and do not provide hand and object poses, which have been shown to be useful cues for a comprehensive understanding of the scene [26, 77].

A few datasets collect hand pose ground truth, acquired in an automated or semi-automated way (Panoptic [37, 38], FreiHand [99], Interhand [55]). However, they do not consider interactions with objects. Recently, GRAB [75] uses a mocap system and objects from [6] to track body and hand pose while interacting with the scene without providing corresponding images. HOnnotate [32] relies on an optimization process to estimate accurate hand and object pose from

Dataset	Frames	Action	6D Obj	3D left	3D right	(**)Markerless	Real	Ego	Depth	Multiview
H2O	571k	✓	✓	✓	✓	✓	✓	✓	✓	✓
FPHA [26]	100k	✓	* (23k)	·	✓	·	✓	✓	✓	·
HOnnotate [32]	78k	·	✓	·	✓	✓	✓	·	✓	✓
Obman [34]	150k	·	✓	·	✓	✓	·	·	✓	·
Freihand [99]	37k	·	·	·	✓	✓	✓	·	·	✓
Panoptic [38]	1.5M	·	·	✓	✓	✓	✓	·	✓	✓
ContactPose [7]	2.9M	·	(***) ✓	✓	✓	·	✓	·	✓	✓

Table 1: A comparison of the existing related *image-based datasets* with 3D annotations for hand interactions. H2O provides a total of 571k frames captured from 5 different views. (*): Object pose provided only for a subset of frames. (**) Methods without markers on hands & objects. (***) Printed, textureless objects.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 14:25:18 UTC from IEEE Xplore. Restrictions apply.

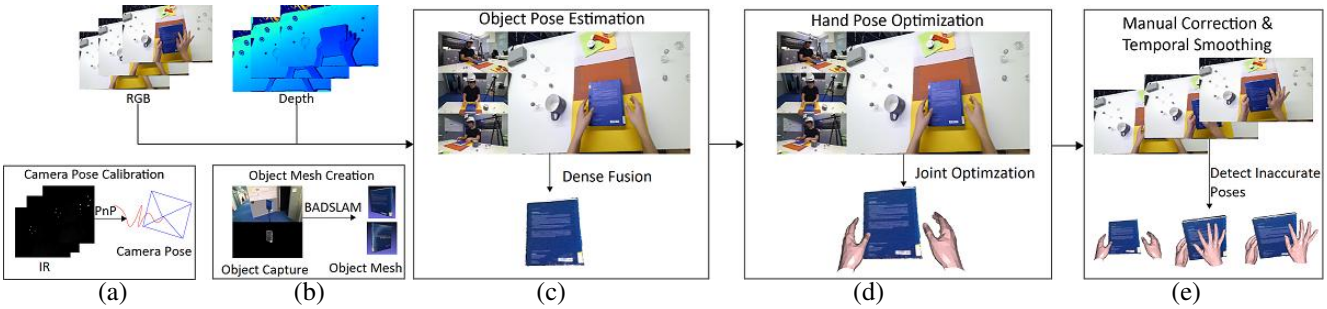


Figure 2: (a) We calibrate cameras using IR sphere markers and PnP [44], (b) create object meshes using BADSLAM [65] on RGB-D captures, and (c) estimate object poses using DenseFusion [83] on RGB-D images and mask images from Mask R-CNN [35]. We then select the pose with the highest confidence among five cameras. (d) Consequently, we detect hand joints with OpenPose [8] and optimize hand shape using Eq. 1. (e) We finally detect and smooth temporally inaccurate poses.

multi-view RGB-D data. ObMan [34] collects purely synthetic images of hands holding objects. All these works, however, consider only single-hand scenarios and do not focus on action recognition. Similarly to us, FPHA [26] collects egocentric RGB-D frames with action, hand and object pose annotations. However, the dataset relies on mangnetic sensors, which pollute the RGB images, and does not include neither multi-view data nor two-hand poses.

As shown in Table 1, our dataset is the first including real, multi-view RGB-D data and accurate annotations for the 3D pose of two hands, object pose, and action labels for egocentric 3D interaction recognition.

Hand & object pose estimation. While a significant amount of research has focused on predicting the pose of hands [27, 54, 56, 58, 70, 93, 94, 98] or objects [5, 48, 59, 78, 83, 90] in isolation, joint understanding of hand-object interactions has received far less attention. Considering hands and objects together adds a number of challenges, which require to reason *e.g.* about occlusions and inter-penetrations. Pioneer works in [1, 79, 80] investigate hand-hand and hand-object interactions relying on optimization frameworks which might be slow and difficult to tune. Tekin et al. [77] and Hasson et al. [33, 34] efficiently estimate hand & object poses directly from RGB images. However they consider only single-hand scenarios.

Recognizing interactions. Action recognition has received a lot of attention in the computer vision community [4, 14, 36, 42, 57, 84, 88]. With the advent of deep learning and the availability of large datasets, significant progress has been made in third-person action recognition [9, 22, 23, 24, 51, 89, 85, 95]. Recently, there has also been an increase in interest for explicitly reasoning about human-object interactions [16, 18, 28, 31, 39, 50, 63, 81, 86, 91, 97] and skeletal action recognition [10, 45, 52, 66, 92], however mostly from third-person viewpoints.

Recognizing interactions from first-person viewpoints, however, poses a number of specific challenges like large occlusions, fast camera motion and background clutter [49]. While initially the lack of large amounts of data somewhat hindered the development of effective DNN-based methods, over the recent years, there has been a renewed interest in the problem. Some approaches leverage multi-modal input like head motion [41, 46, 64, 72] and eye gaze [20, 46]. It is also common to extract features with CNNs and leverage additional 2D cues related to motion, hand location, object location or object class, in isolation [3, 46, 53] or jointly [19, 21, 25, 67, 74]. While all these methods focus on 2D features, recent work [26, 77] suggests that 3D cues (like hand and object pose) can be effective in the context of egocentric action recognition. However, existing methods have focused on single-hand tracking and no attention has been paid so far to estimating the pose of two hands interacting with objects – a scenario which is more representative of the interactions encountered in real-world scenarios.

3. Annotation Method

Fig. 2 shows an overview of our annotation pipeline. We capture synchronized RGB-D frames from multiple views with five Azure Kinect cameras [76]. One of the cameras is mounted on a helmet worn by different subjects to capture egocentric frames. We acquire ground-truth hand and object poses in a semi-automated way. First, we scan each object with a Kinect to obtain a complete 3D model. This model is used to track object 6D pose in each frame via DenseFusion [83]. To track hands, we fit the MANO parametric hand model [62] to multi-view depth data in each frame. This automated tracking process may fail on some frames, due to challenges like (self-)occlusions, blur and cluttered background. We therefore manually detect failure cases and remove corresponding poses; such poses can be then replaced via temporal smoothing. Finally, we manually annotate action labels over the sequences. In the following sections, we describe each step of our pipeline in detail.

3.1. Camera Calibration

Our setup consists of four static plus one head-mounted RGB-D cameras. We use the factory-calibrated intrinsic parameters accessible via Azure Kinect DK [76]. As for the extrinsic parameters, we obtain them with a calibration method relying on IR reflective spheres. We choose this method to make our setup portable and easy to deploy.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 14:25:18 UTC from IEEE Xplore. Restrictions apply.

We place nine IR reflective spheres at random locations in the scene, ensuring that each sphere is visible from all the cameras. In the IR images captured by our cameras to reconstruct depth, such spheres are shown as bright circles, which can be easily detected in an automated way. We compute the center of each sphere and then obtain its 3D location by considering the corresponding pixel in the depth image. Given the 3D location of the nine spheres in each frame, we solve for camera pose via PnP [43]. In order to consistently identify spheres across frames, we define an initial mapping in the first frame and then track it over time.

Poses computed for the head-mounted camera can exhibit jitter. We smooth them via Kalman filtering [82], under the assumption that the head moves with uniform speed. The overall framework allows us to use multiple cameras during annotation, which eventually increases the fidelity and accuracy of our annotations.

3.2. Object Pose Annotation

We obtain accurate per-frame object 6D poses using multi-view images together with camera pose information. We first reconstruct a 3D mesh model for each object. To this end, we scan the object by capturing RGB-D frames with a hand-held Kinect camera moving around it. We feed these frames into a state-of-the-art RGB-D SLAM method, BADSLAM [65], to reconstruct a 3D mesh. We obtain texture for each object in Blender [11]: we project the RGB images obtained at scanning time onto the mesh surface, using the camera pose returned by BADSLAM.

We leverage these models to train an object pose tracker. First, we train an object mask predictor based on Mask R-CNN [35]. As training data, we use the masks obtained by projecting our 3D models onto the images used for their BADSLAM-based reconstruction. Then, we feed mask predictions together with the corresponding RGB-D images into DenseFusion [83] to estimate object pose. We obtain a pose prediction for each camera view, and select the one with the highest confidence. Finally, we refine this pose estimate via ICP [43]. Namely, we compute a point cloud from each of the five depth images and merge them into a single point cloud by using camera pose information; then, we fit our object model to this point cloud, taking the prediction from DenseFusion as initialization.

3.3. Hand Pose Annotation

For hand pose estimation, we rely on the widely used MANO hand model [62]. MANO factorizes human hand shape into a set of identity parameters $\beta \in \mathbb{R}^{10}$ and a set of pose parameters $\theta \in \mathbb{R}^{51}$, storing angles for 15 skeleton joints plus global rotation and translation. Formally, we can define MANO as a function $H_V(\theta, \beta)$ returning a triangulated mesh with N_V vertices. We also define the MANO skeleton as a function $H_J(\theta, \beta)$ which returns $N_J = 21$ joint locations (the 15 original ones plus 6 other for fingertips and wrist, to map to the OpenPose [8] skeleton – see Supp. Mat.). We take the object pose estimated as above, and we leverage it when tracking hand pose.

We track hands by minimizing at each frame, f, a loss function defined as:

$$\hat{\theta}_f = \underset{\theta}{\operatorname{argmin}} \sum_{c=1}^{N_C} (\lambda_1 \mathcal{L}_s + \lambda_2 \mathcal{L}_{2D}) + \lambda_3 \mathcal{L}_{3D} + \lambda_4 \mathcal{L}_p + \lambda_5 \mathcal{L}_{phy} + \lambda_6 \mathcal{L}_a + \lambda_7 \mathcal{L}_m \tag{1}$$

where N_c is the number of cameras. Here, $\mathcal{L}_s$ is a silhouette-based error term, $\mathcal{L}_{2D}$ and $\mathcal{L}_{3D}$ measure joint error in 2D and 3D, respectively, $\mathcal{L}_p$ and $\mathcal{L}_a$ are regularizers for pose, $\mathcal{L}_{phy}$ penalizes physically implausible interpenetrations between hand and objects, and $\mathcal{L}_m$ penalizes distance in 3D between the hand depth data and the MANO surface. Lambdas weight the contribution of each error term. Note that, in order to obtain subject-specific parameters, we minimize Eq. (1) with respect to $\boldsymbol{\beta}$ on one frame only. Then, we track hand pose over the sequence by keeping β fixed and optimizing Eq. (1) with respect to θ (for both left and right hands). We omit β from the following equations for simplicity.

2D joint error. We penalize distance in 2D between MANO joints and OpenPose estimates by defining:

$$\mathcal{L}_{2D}(\theta) = \sum_{i=1}^{N_J} \| J_{2D,c}[i] - \Pi_c(H_J(\theta)[i]) \| \tag{2}$$

where J_{2D} denotes the 2D joint positions pre-computed with OpenPose, and $H_J(\theta)[i]$ returns the ith 3D joint location of the MANO skeleton.

3D joint error. Similarly to the 2D joint error, we compute a penalty in 3D by triangulating OpenPose estimates. We found that using this error term helps achieve faster convergence and increase stability.

3D mesh surface error. We obtain a point cloud for hand data by merging the point clouds obtained from each depth image across our different views, and segmenting out the points that do not project onto the hand mask computed as above. Our 3D surface error term penalizes the distance between this point cloud and the MANO surface:

$$\mathcal{L}_m(\theta) = \sum_{i=1}^{N_V} \left\| (p_j - H_V(\theta)[i]) \cdot H_V^{\perp}(\theta)[i] \right\| \quad \text{where } j = \underset{j}{\operatorname{argmin}} \| p_j - H_V(\theta)[i] \| \tag{3}$$

where p_j is the jth point of the point cloud, and $H_V^{\perp}(\theta)$ denotes the normal of the hand mesh vertex i.

As shown in Eq. 1, our optimization function further includes a silhouette error term and regularizers for joint angle limits and physical constraints. We refer the reader to the Supp. Mat. for more details on these terms and for an

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 14:25:18 UTC from IEEE Xplore. Restrictions apply.

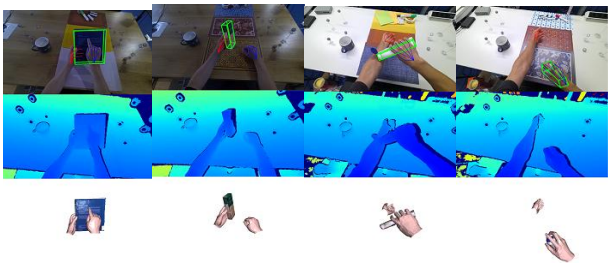


Figure 3: RGB and depth images with the corresponding annotations of hand & object pose, and action label. First row: Left hand keypoints, right hand keypoints, and 3D object bounding box are projected on the RGB image. Second row: Synchronous depth images. Third row: Ground-truth data for hand and object meshes. We provide further examples of ground-truth data in Supp. Mat.

ablation study on the influence of different error terms in the annotation accuracy.

After running our automated pipeline, we inspect all the frames to identify and remove inaccurate poses. As a final step, we smooth and interpolate poses via Kalman filtering.

3.4. Temporal Action Annotation

We provide action labels as verb-noun pairs. We consider 11 verb classes: *grab*, *place*, *open*, *close*, *pour*, *take out*, *put in*, *apply*, *read*, *spray* and *squeeze*. As for nouns, we consider 8 classes: *book*, *espresso*, *lotion*, *spray*, *milk*, *cocoa*, *chips*, *cappuccino*. By combining verbs and nouns, after excluding pairs which are not represented in our dataset, we obtain a total of 36 action classes. Note that we pick only one verb and one noun for every frame, so there are no overlapping action labels. We select action labels for the entire dataset manually, using the VIA annotation tool [15]. Fig. 3 shows some annotation examples.

4. The H2O Dataset

We acquired the images of the H2O dataset in indoor settings in which the subjects interact with eight different objects using both of their hands. The dataset includes 571,645 RGBD frames, and features four participants performing 36 distinct action classes in three different environments. With the methodology described in Sec. 3, we annotate accurate ground-truth data for left and right hand pose, 6D object pose, camera pose and action labels. In our dataset we further provide MANO [62] hand fits for both left and right hand, and high-quality object meshes. In addition, we also compute scene point clouds using the camera poses and the synchronized RGBD data. Altogether, the curated dataset allows for a comprehensive understanding of the egocentric scene.

4.1. Capture Setup

Fig. 4(c) demonstrates our data capture setup. We use five Azure Kinect cameras to acquire synchronized RGB and depth images. To ensure synchronization between multiple cameras, we use physical cables between them. This results in less than 100 microseconds of lag between cameras [76]. As instructed in [76], to avoid interference between multiple depth cameras, we further offset camera captures from one another by 160 microseconds, which results in a total maximum of only 0.74 ms of delay between cameras. We place four different static cameras at arbitrary locations that cover hand-object interactions. An egocentric camera is further mounted on the forehead of a helmet and adjusted by the participants to set egocentric views. We calibrate all of the five cameras with nine IR reflective balls as explained in Sec. 3.1. The data is acquired in three environments (*e.g.* hall, office and kitchen) using several different backgrounds. We record videos at a resolution of 1280x720 pixels for both RGB and depth images with a frame rate of 30 fps. Each video corresponds to a series of actions involving various hand-object interactions.

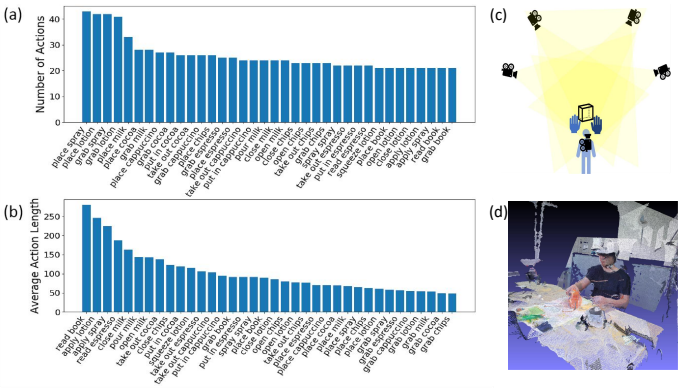


Figure 4: (a) Number of instances per action in the H2O dataset. (b) Average number of frames per each action class. (c) Schematic camera capture setup. Four static cameras can capture parts that can not be observed by the egocentric view. (d) Scene point cloud computed from multi-view data.

4.2. Dataset Statistics

We divide the dataset into a training and test set. We split the training and test data with a subject-based split where we leave one subject out for testing and the rest for training. We further use a part of the training data of one subject as the validation dataset for model selection. The data from multiple views consists of 344,645 frames for training, 73,380 frames for validation and 153,620 frames for testing.

We plot the number of instances per action and the average number of frames for each action class in Fig. 4. Action instances are well distributed across the dataset with the least frequent action appearing 21 times. In the dataset, both hands are used in 57.8%, only left hand in 12.4%, and only right hand in 29.8% of the dataset. The length of action clips spans a wide range demonstrating the diversity of the dataset that includes both slow and fast actions.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 14:25:18 UTC from IEEE Xplore. Restrictions apply.

5. Recognizing 3D Hand-Object Interactions

Given the rich annotations of H2O, our goal is to construct comprehensive interpretations of egocentric scenes from image sequences to understand human interactions. For this purpose, we propose a unified framework that jointly estimates the poses of two hands & the manipulated objects, and recognizes egocentric interactions. We use this framework to establish baselines on first person interaction recognition and hand & object pose estimation.

Pose Prediction. We build upon the network architecture of [77] to estimate the poses of both left and right hand, and the pose of the manipulated object. While [77] addresses only single hand scenarios, in our case, we aim to predict the pose of both hands. To this end, each frame in a sequence is passed through a fully convolutional network with a backbone of YOLOv2 [61]. We produce a 3D grid as the output of our fully convolutional network, instead of producing a 2D grid as in [61]. To be able to predict the pose of both hands and objects at the same time, we associate each output grid cell with 3 vectors for left hand, right hand and the manipulated object. These vectors contain target values for left hand ($\mathbf{y}_i^{h,l}$), right hand ($\mathbf{y}_i^{h,r}$) and object pose ($\mathbf{y}_i^{o}$), with overall confidence values ($c^{h,l}$,$c^{h,r}$, c^{o}) for individual pose predictions. The confidence values are defined on-the-fly during training as a function of the distance of the predicted poses to the ground-truth ones. The final layer of our single-shot network produces, for each cell i, predictions for left hand ($\hat{\mathbf{y}}_i^{h,l}$), right hand ($\hat{\mathbf{y}}_i^{h,r}$) and object ($\hat{\mathbf{y}}_i^{o}$), along with their associated overall confidence values, $\hat{c}_i^{h,l}$, $\hat{c}_i^{h,r}$ and $\hat{c}_i^{o}$. For each frame, the loss function to train our network is defined as follows:

$$\mathcal{L} = \lambda_{pose} \sum_i (||\hat{\mathbf{y}}_i^{h,l} - \mathbf{y}_i^{h,l}|| + ||\hat{\mathbf{y}}_i^{h,r} - \mathbf{y}_i^{h,r}|| + ||\hat{\mathbf{y}}_i^{o} - \mathbf{y}_i^{o}||) \quad (4)$$

$$+ \lambda_{conf} \sum_i ((\hat{c}_i^{h,l} - c_i^{h,l})^2 + (\hat{c}_i^{h,r} - c_i^{h,r})^2 + (\hat{c}_i^{o} - c_i^{o})^2) \quad (5)$$

While the poses for the left and right hand are defined by 3D joint coordinates, object pose is parameterized by corner points of a 3D bounding box surrounding the object. Given the control point predictions of the network on the 3D bounding box, 6D object pose can be efficiently computed by aligning the predictions to the reference 3D bounding box with a rigid transformation. Predictions with low confidence values are pruned and the ones with high confidence values are selected as pose predictions.

Interaction Recognition. RNNs have been successfully used before to recognize actions [2, 77]. However they do not fully leverage the special graph structure of the skeleton data for hand-object interactions. Therefore, we resort to parameterizing the left hand skeleton, right hand skeleton and object bounding box as individual graphs and combine them in a multi-graph structure. We then compute the topology of the multi-graph structure using a graph convolutional network (GCN) by learning the links across hand and object locations that are involved in interaction. While modeling intra-dependencies within a single graph, this framework also allows for learning interdependencies between left hand-right hand, left hand-object, and right hand-object.

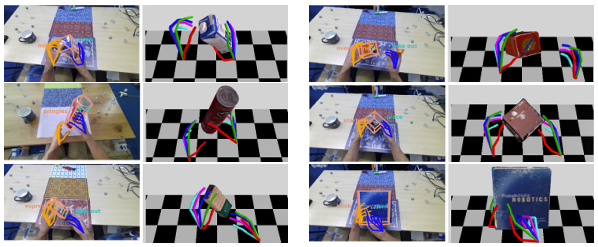


Figure 5: Qualitative results on the $H2O$ dataset. We show estimated hand 3D pose, object 6D pose, and action labels. The proposed method can properly handle challenging occlusions.

More particularly, we employ a spatiotemporal graph to encode both spatial and temporal information as in ST-GCN [92] and 2s-AGCN [66]. Standard ST-GCN [92] for human action recognition models structured information between body skeleton joints using

$$\mathbf{f_{out}} = \sum_j \mathbf{W_j} \mathbf{f_{in}} (\mathbf{A_j} \odot \mathbf{M_j}) \quad (6)$$

where $\mathbf{f_{in}} \in \mathbb{R}^{C_{in} \times T \times N}$ is an input feature map, $\mathbf{A_j} \in \mathbb{R}^{N \times N}$ is an adjacency matrix that represents skeletal connections, $\mathbf{W_j} \in \mathbb{R}^{C_{out} \times C_{in} \times 1 \times 1}$ is a weight vector of 1×1 convolutions and $\mathbf{M_j} \in \mathbb{R}^{N \times N}$ is an attention map. Here j denotes the vertex neighborhood defined by the convolutional kernel, C is the number of channels, T is the temporal length and N is the number of vertices. ST-GCN works on a single graph entity, *e.g.* human skeleton, and models intra-skeleton connections with a fixed adjacency matrix. In our case, in addition to intra-graph dependencies, we aim to model also inter-graph dependencies between hands and objects. Since each time different hand and object parts are involved in interactions, a fixed adjacency matrix to model inter-dependencies would not yield optimal results. Therefore, individually for left hand, right hand and object, we employ the following to be able to model their dependencies:

$$\mathbf{f_{out}} = \sum_j \mathbf{W_j} \mathbf{f_{in}} (\mathbf{A_{j,intra}} + \mathbf{A_{j,inter}} + \mathbf{T_{j,intra}} + \mathbf{T_{j,inter}} + \mathbf{S_j}) \quad (7)$$

While $\mathbf{A_{j,intra}}$ plays the same role as $\mathbf{A_j}$ in Eq. 6 for left hand, right hand and object, $\mathbf{A_{j,inter}}$ models interrelated dependencies between hands and objects via static connections between symmetric hand parts and object center. Here, both of these matrices are fixed adjacency matrices as in ST-GCN (Eq. 6). In addition to them, we represent inter-connections between left hand and right hand, left hand and object, and right hand and object with an additional adjacency matrix, $\mathbf{T_{j,inter}}$. Differently from $\mathbf{A_{j,inter}}$, $\mathbf{T_{j,inter}}$ is not fixed, but rather parameterized. Its values are unconstrained and jointly optimized with other

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 14:25:18 UTC from IEEE Xplore. Restrictions apply.

network parameters, which means that the graph topology and edge weights are fully learned from the training data. In addition to $\mathbf{T_{j,inter}}$, we also use an additional parameterized adjacency matrix, $\mathbf{T_{j,intra}}$ that adaptively learns intra-related dependencies within single graph entities (*e.g.* left hand, right hand or object) during interaction. This data-driven model allows us to learn graphs that are fully targeting the hand-object interaction task.

Note that in contrast to Eq. 6, we do not use an attention map as in [66], since our *parameterized* adjacency matrices can play the same role of the attention mechanism performed by $\mathbf{M_j}$ in Eq. 6 to attribute more importance to edges between hands and objects that are involved in interaction. Besides in Eq. 6, if one of the elements of $\mathbf{A_j}$ is 0, the result will be 0 regardless of the value of $\mathbf{M_j}$ due to the dot multiplication. Therefore we use addition instead of dot multiplication in Eq. 7 to allow for forming new connections between our graphs. Similarly with [66], we use an additional data-dependent term, $\mathbf{S_j}$ in our formulation which learns a unique graph for each sample that use the dot product to measure the similarity of the two vertices in an embedding space.

$$\mathbf{S_j} = softmax(\mathbf{f_{in}^T W_{\theta j}^T W_{\phi j} f_{in}}) \tag{8}$$

where $\mathbf{W}_\theta$ and $\mathbf{W}_\phi$ are the parameters of the embedding functions θ and ϕ, respectively. Here, embedding functions are chosen as 1×1 convolutional layers.

By stacking the layers defined by Eq. 7, with a total of 10 layers, we build our topology-aware graph convolutional network (TA-GCN) for 3D interaction recognition. It takes at each iteration the combination of $\hat{\mathbf{y}}_i^{h,l}$, $\hat{\mathbf{y}}_i^{h,r}$ and $\hat{\mathbf{y}}_i^{o}$ as its initial feature map to model hand object-interactions. We demonstrate learned graph connections for a hand-object interaction scenario in Fig. 8 and analyze our design choices in Sec. 6. We provide further details for the architecture, hyperparameters and training of the pose prediction and interaction recognition models in the Supp. Mat..

6. Evaluation

In this section, we first verify the accuracy of our ground-truth annotations. We then present baseline results on hand & object pose estimation and egocentric action recognition on our dataset. For the latter, we also compare our baseline approach against the state-of-the-art in action recognition and demonstrate the clear benefits of our approach based on hand-object poses with respect to the existing methods.

6.1. Dataset Analysis

Verfication. We verify the accuracy of our hand-object pose annotations on a random split of our dataset. To that end, we annotate 500 images on 5 different camera views with the fingertips of the hand and the predefined keypoints of the manipulated objects. We then triangulate these 2D points to get manual 3D annotations for hands and objects. We compute the distance of our annotations to those of the manually created ones to measure the accuracy of our poses. We demonstrate the results of our verification in Table 2. For both hands and the object, the error is approximately within a range of 1 cm, which demonstrates the high precision of our dataset. Our error margin is comparable with those of [32, 99] even though our dataset features more mutual occlusions due to two-hand manipulation.

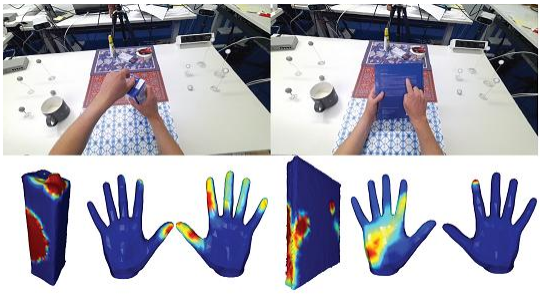


Figure 6: Contact modelling on H2O. Our dataset facilitates modelling hand-object contact and 3D affordances.

Pose feature	Object	Left hand	Right hand
Mean (std)	1.10 (±0.37)	0.82 (±0.43)	0.93 (±0.57)

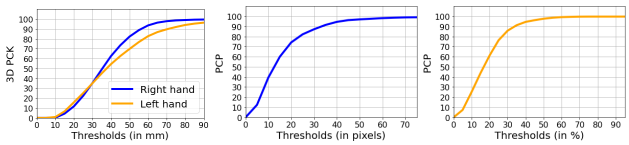
Table 2: Hand & object pose verification results (in cm) for evaluating the accuracy of the provided ground-truth data.

Contact Modelling. Having precise hand & object pose annotations and meshes, H2O further facilitates modelling hand-object contact [7, 75]. To this end, for each vertex in the hand mesh, we find the nearest vertices on the object within a certain a threshold (*e.g.* 2 cm). We then compute a histogram counting the number of neighbors for each vertex of the MANO mesh and normalize it to model contact hotspots on hand. We repeat the same procedure also for the object mesh to create a contact map on the object surface. We visualize example contact maps of our dataset in Fig. 6.

6.2. Experimental Results

Predicting jointly the 3D pose of two hands and the manipulated objects. We train and evaluate our method using the training, validation and test splits described in Sec. 4.2 and report baseline pose estimation accuracies for hands and objects in Fig. 7. We use the percentage of correctly estimated poses to evaluate hand and object pose estimation accuracy. Specifically, we use the 3D PCK metric for hand pose estimation, and the 2D reprojection and ADD metrics for object pose estimation as in [77]. We demonstrate that our method can reliably predict the pose of two hands and the manipulated objects with a low error margin and constitutes a strong baseline for joint pose estimation of two hands interacting with objects. Note also that our approach constitutes the first method and baseline for estimating the pose of *two hands interacting with objects* from a single RGB image. We still evaluate our approach against single

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 14:25:18 UTC from IEEE Xplore. Restrictions apply.



(a) (b) (c)

Figure 7: Pose estimation results on the H2O dataset using different thresholds, for (a) hands with 3D PCK metric, and for objects with (b) 2D reprojection and (c) ADD metric.

Model	Acc. (%)
LEFT HAND	33.61
OBJECT	48.55
RIGHT HAND	52.70
BOTH HANDS	58.92
ALL	79.25

(a)

Model	Acc. (%)
NO INTERCONNECTION	75.52
LEFT HAND-RIGHT HAND	76.76
HANDS-OBJECT	78.84
ALL INTERCONNECTIONS	79.25

(b)

Model	Acc. (%)
ST-GCN	73.86
TA-GCN wo $\mathbf{S_j}$	73.44
TA-GCN wo $\mathbf{T_{j,inter}}$	75.52
TA-GCN wo $\mathbf{T_{j,intra}}$	76.76
TA-GCN wo $\mathbf{A_{j,inter}}$	76.35
TA-GCN wo $\mathbf{A_{j,intra}}$	77.59
TA-GCN	79.25

(c)

Table 3: Impact of different (a) input modalities, (b) interconnections and (c) graph terms on interaction recognition accuracy.

hand-object pose estimation methods of [33, 77], for comparison purposes, in Table 4 and further provide qualitative examples of our pose predictions in Fig. 5.

Interaction recognition. In Table 3(a), we show the influence of different input modalities on the accuracy of interaction recognition on the H2O dataset. To this end, we evaluate the impact of hand & object poses for interaction recognition. Hand pose and object keypoints are predicted through our single pass network described in Sec. 5. We show that the combination of right and left hand pose as well as the combination of hand and object poses significantly improve overall action recognition scores, which demonstrates the individual contributions and the complementary nature of each input modality. We further evaluate the importance of modelling inter-dependencies between both hands and objects in Table 3(b) and demonstrate that modelling interdependencies between left hand & right hand, and hands & objects, boosts the accuracy for recognizing interactions. In Table 3(c), we evaluate the influence of different terms of Eq. 7 and demonstrate that with all the graphs added together, the model obtains the best results compared to the baselines. We visualize the learned connections of our model in Fig. 8.

We further compare our action recognition accuracy to the state-of-the-art image-based learning methods of C2D [87], I3D [9] and SlowFast [23] using the PySlowFast library [17] and pose-based learning methods of H+O [77] and ST-GCN [92] and show our results in Table 4. Following [17], we train image-based models using a batch size of 16 and a temporal window size of 64 frames with a sampling ratio of 2. We use a ResNet-50 backbone and train the network using SGD with a learning rate of 0.1. Pose-based methods are trained as in [77, 92], and evaluated with the estimated poses using our method from RGB images of our dataset. Our approach to interaction recognition achieves the highest validation and test accuracy on the H2O dataset,

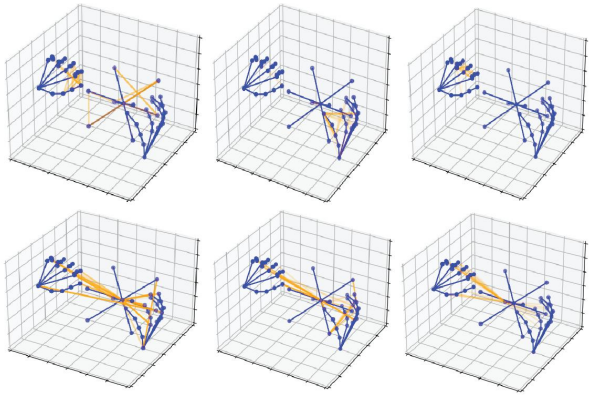


Figure 8: Learned graph connections for different layers. We demonstrate the top-20 learned intra-(top) and inter-(bottom) connections at layer 1, 5 and 9, respectively, in each column. The thickness of the connections corresponds to the weight of learned connection value. Hand-object connections are given more weight than hand-hand connections during interaction with an object. Our model attributes more importance to fingertips and DIP joints that are more commonly involved in manipulation.

Method	Left h.	Right h.	Object
Hasson[33]	**39.56**	-	67.47
Hasson[33]	-	41.87	66.05
H+O[77]	41.42	-	48.06
H+O[77]	-	38.86	52.57
Ours	41.45	**37.21**	**47.90**

Model	Val acc. (%)	Test acc. (%)
C2D [87]	76.10	70.66
I3D [9]	85.15	75.21
SlowFast [23]	86.00	77.69
H+O [77]	80.49	68.88
ST-GCN [92]	83.47	73.86
OURS (TA-GCN)	**86.78**	**79.25**

Table 4: Pose errors (left, in mm) and action accuracies (right). Single hand methods of [33, 77] are separately trained for left & right hand. [77, 92] use pose predictions of our method.

demonstrating the effectiveness of our method and the importance of the 3D pose predictions facilitated by H2O.

7. Conclusion

In this paper, we propose a method to collect a dataset of two hands manipulating objects for first person interaction recognition. We provide a rich set of annotations including action labels, object classes, 3D left & right hand poses, 6D object poses, camera poses and scene point clouds. We further propose the first method to jointly recognize the 3D poses of two hands manipulating objects and a novel topology-aware graph convolutional network for recognizing hand-object interactions. Our framework models the interactions between hands and objects in 3D to recognize actions from first-person views and yields state-of-the-art accuracy. We believe that our dataset and experiments can be of interest to communities of 3D hand pose estimation, 6D object pose estimation, hand-object interaction, robotics and action recognition, and help bridge the gap between hand-object interaction and egocentric action recognition.

Acknowledgements. Taein Kwon was supported by the Microsoft Mixed Reality & AI Zürich Lab PhD scholarship. The authors thank Silvano Galliani, Joshua Elsdon, Yana Hasson, Jeff Delmerico, Helen Oleynikova and Mihai Dusmanu for helpful discussions.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 14:25:18 UTC from IEEE Xplore. Restrictions apply.

References

[1] Luca Ballan, Aparna Taneja, Jürgen Gall, Luc Van Gool, and Marc Pollefeys. Motion capture of hands in action using discriminative salient points. In *ECCV*, 2012. 3

[2] Fabien Baradel, Natalia Neverova, Christian Wolf, Julien Mille, and Greg Mori. Object level visual reasoning in videos. In *ECCV*, 2018. 6

[3] Gedas Bertasius, Hyun Soo Park, Stella X Yu, and Jianbo Shi. First person action-object detection with egonet. *arXiv preprint arXiv:1603.04908*, 2016. 3

[4] Aaron F. Bobick and James W. Davis. The recognition of human movement using temporal templates. *PAMI*, 23(3):257–267, 2001. 3

[5] Eric Brachmann, Frank Michel, Alexander Krull, Michael Ying Yang, Stefan Gumhold, et al. Uncertainty-driven 6d pose estimation of objects and scenes from a single rgb image. In *CVPR*, 2016. 3

[6] Samarth Brahmbhatt, Cusuh Ham, Charles C. Kemp, and James Hays. ContactDB: Analyzing and predicting grasp contact via thermal imaging. In *CVPR*, 2019. 2

[7] Samarth Brahmbhatt, Chengcheng Tang, Christopher D. Twigg, Charles C. Kemp, and James Hays. ContactPose: A dataset of grasps with object contact and hand pose. In *ECCV*, 2020. 2, 7

[8] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *PAMI*, 43(1):172–186, 2019. 3, 4

[9] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017. 1, 2, 3, 8

[10] Ke Cheng, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with shift graph convolutional network. In *CVPR*, 2020. 3

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

[12] Dima Damen, Hazel Doughty, Giovanni Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Computer Architecture Letters*, (01):1–1, 2020. 2

[13] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. In *ECCV*, 2018. 2

[14] Piotr Dollár, Vincent Rabaud, Garrison Cottrell, and Serge Belongie. Behavior recognition via sparse spatio-temporal features. In *International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 2005. 3

[15] Abhishek Dutta and Andrew Zisserman. The VIA annotation software for images, audio and video. In *Proceedings of the 27th ACM International Conference on Multimedia*, 2019. 5

[16] Victor Escorcia and Juan Niebles. Spatio-temporal human-object interactions for action recognition in videos. In *ICCV Workshops*, 2013. 3

[17] Haoqi Fan, Yanghao Li, Bo Xiong, Wan-Yen Lo, and Christoph Feichtenhofer. Pyslowfast. https://github.com/facebookresearch/slowfast, 2020. 2, 8

[18] Hao-Shu Fang, Jinkun Cao, Yu-Wing Tai, and Cewu Lu. Pairwise body-part attention for recognizing human-object interactions. In *ECCV*, 2018. 3

[19] Alireza Fathi, Ali Farhadi, and James M Rehg. Understanding egocentric activities. In *ICCV*, 2011. 3

[20] Alireza Fathi, Yin Li, and James M Rehg. Learning to recognize daily actions using gaze. In *ECCV*, 2012. 3

[21] Alireza Fathi, Xiaofeng Ren, and James M Rehg. Learning to recognize objects in egocentric activities. In *CVPR*, 2011. 3

[22] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *CVPR*, 2020. 1, 3

[23] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019. 1, 2, 3, 8

[24] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *CVPR*, 2016. 1, 3

[25] David F Fouhey, Wei-cheng Kuo, Alexei A Efros, and Jitendra Malik. From lifestyle vlogs to everyday interactions. In *CVPR*, 2018. 3

[26] Guillermo Garcia-Hernando, Shanxin Yuan, Seungryul Baek, and Tae-Kyun Kim. First-person hand action benchmark with rgb-d videos and 3d hand pose annotations. In *CVPR*, 2018. 1, 2, 3

[27] Liuhao Ge, Yujun Cai, Junwu Weng, and Junsong Yuan. Hand pointnet: 3d hand pose estimation using point sets. In *CVPR*, 2018. 3

[28] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object interactions. In *CVPR*, 2018. 3

[29] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *ICCV*, 2017. 2

[30] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *CVPR*, 2018. 2

[31] Tanmay Gupta, Alexander Schwing, and Derek Hoiem. No-frills human-object interaction detection: Factorization, layout encodings, and training techniques. In *ICCV*, 2019. 3

[32] Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. Honnotate: A method for 3d annotation of hand and object poses. In *CVPR*, 2020. 1, 2, 7

[33] Yana Hasson, Bugra Tekin, Federica Bogo, Ivan Laptev, Marc Pollefeys, and Cordelia Schmid. Leveraging photometric consistency over time for sparsely supervised hand-object reconstruction. In *CVPR*, pages 571–580, 2020. 3, 8

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 14:25:18 UTC from IEEE Xplore. Restrictions apply.

[34] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *CVPR*, 2019. 1, 2, 3

[35] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 3, 4

[36] Hueihan Jhuang, Thomas Serre, Lior Wolf, and Tomaso Poggio. A biologically inspired system for action recognition. In *ICCV*, 2007. 3

[37] Hanbyul Joo, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *ICCV*, 2015. 2

[38] Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Scott Godisart, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social interaction capture. *PAMI*, 41(1):190–204, 2017. 2

[39] Keizo Kato, Yin Li, and Abhinav Gupta. Compositional learning for human object interaction. In *ECCV*, 2018. 3

[40] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 2

[41] Kris M Kitani, Takahiro Okabe, Yoichi Sato, and Akihiro Sugimoto. Fast unsupervised ego-action learning for first-person sports videos. In *CVPR*, 2011. 3

[42] Ivan Laptev. On space-time interest points. *IJCV*, 64(2-3):107–123, 2005. 3

[43] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o (n) solution to the pnp problem. *IJCV*, 81(2):155, 2009. 4

[44] V. Lepetit, F. Moreno-Noguer, and P. Fua. EPnP: An Accurate O(n) Solution to the PnP Problem. *IJCV*, 81(2):155–166, 2009. 3

[45] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *CVPR*, 2019. 3

[46] Yin Li, Alireza Fathi, and James M Rehg. Learning to predict gaze in egocentric video. In *ICCV*, 2013. 3

[47] Yin Li, Miao Liu, and James M Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. In *ECCV*, 2018. 2

[48] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. Deepim: Deep iterative matching for 6d pose estimation. In *ECCV*, 2018. 3

[49] Yin Li, Zhefan Ye, and James M Rehg. Delving into egocentric actions. In *CVPR*, 2015. 1, 3

[50] Yong-Lu Li, Siyuan Zhou, Xijie Huang, Liang Xu, Ze Ma, Hao-Shu Fang, Yanfeng Wang, and Cewu Lu. Transferable interactiveness knowledge for human-object interaction detection. In *CVPR*, 2019. 3

[51] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *ICCV*, 2019. 3

[52] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *CVPR*, 2020. 3

[53] Minghuang Ma, Haoqi Fan, and Kris M Kitani. Going deeper into first-person activity recognition. In *CVPR*, 2016. 3

[54] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. V2v-posenet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map. In *CVPR*, 2018. 3

[55] Gyeongsik Moon, Shoou-I Yu, He Wen, Takaaki Shiratori, and Kyoung Mu Lee. Interhand2. 6m: A dataset and baseline for 3d interacting hand pose estimation from a single rgb image. *arXiv preprint arXiv:2008.09309*, 2020. 2

[56] Franziska Mueller, Dushyant Mehta, Oleksandr Sotnychenko, Srinath Sridhar, Dan Casas, and Christian Theobalt. Real-time hand tracking under occlusion from an egocentric rgb-d sensor. In *ICCVW*, 2017. 3

[57] Juan Carlos Niebles, Hongcheng Wang, and Li Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *IJCV*, 79(3):299–318, 2008. 3

[58] Markus Oberweger and Vincent Lepetit. Deepprior++: Improving fast and accurate 3d hand pose estimation. In *ICCVW*, 2017. 3

[59] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *CVPR*, 2019. 3

[60] Francesco Ragusa, Antonino Furnari, Salvatore Livatino, and Giovanni Maria Farinella. The meccano dataset: Understanding human-object interactions from egocentric videos in an industrial-like domain. *arXiv preprint arXiv:2010.05654*, 2020. 2

[61] J. Redmon and A. Farhadi. YOLO9000: Better, Faster, Stronger. In *CVPR*, 2017. 6

[62] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (ToG)*, 36(6):245, 2017. 3, 4, 5

[63] Amir Rosenfeld and Shimon Ullman. Hand-object interaction and precise localization in transitive action recognition. In *2016 13th Conference on Computer and Robot Vision (CRV)*, pages 148–155. IEEE, 2016. 3

[64] Michael S Ryoo and Larry Matthies. First-person activity recognition: What are they doing to me? In *CVPR*, 2013. 3

[65] Thomas Schops, Torsten Sattler, and Marc Pollefeys. Bad slam: Bundle adjusted direct rgb-d slam. In *CVPR*, pages 134–144, 2019. 3, 4

[66] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *CVPR*, 2019. 3, 6, 7

[67] Gunnar A Sigurdsson, Abhinav Gupta, Cordelia Schmid, Ali Farhadi, and Karteek Alahari. Actor and observer: Joint modeling of first and third-person videos. In *CVPR*, 2018. 3

[68] Gunnar A Sigurdsson, Abhinav Gupta, Cordelia Schmid, Ali Farhadi, and Karteek Alahari. Charades-ego: A large-scale dataset of paired third and first person videos. *arXiv preprint arXiv:1804.09626*, 2018. 2

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 14:25:18 UTC from IEEE Xplore. Restrictions apply.

[69] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *ECCV*. Springer, 2016. 2

[70] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *CVPR*, 2017. 3

[71] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, 2014. 1

[72] Suriya Singh, Chetan Arora, and CV Jawahar. First person action recognition using deep learned descriptors. In *CVPR*, 2016. 3

[73] Srinath Sridhar, Franziska Mueller, Michael Zollhoefer, Dan Casas, Antti Oulasvirta, and Christian Theobalt. Real-time joint tracking of a hand manipulating an object from rgb-d input. In *ECCV*, 2016. 1

[74] Sudeep Sundaram and Walterio W Mayol Cuevas. High level activity recognition using low resolution wearable vision. In *CVPRW*, 2009. 3

[75] Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas. GRAB: A dataset of whole-body human grasping of objects. In *ECCV*, 2020. 2, 7

[76] Azure SDK Team. *Azure Kinect SDK*. Microsoft, 2020. 3, 5

[77] Bugra Tekin, Federica Bogo, and Marc Pollefeys. H+ o: Unified egocentric recognition of 3d hand-object poses and interactions. In *CVPR*, 2019. 2, 3, 6, 7, 8

[78] Bugra Tekin, Sudipta N Sinha, and Pascal Fua. Real-time seamless single shot 6d object pose prediction. In *CVPR*, 2018. 3

[79] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. Capturing hands in action using discriminative salient pointsand physics simulation. *IJCV*, 2016. 2, 3

[80] Dimitrios Tzionas, Abhilash Srikantha, Pablo Aponte, and Juergen Gall. Capturing hand motion with an rgb-d sensor, fusing a generative model with salient points. In *GCPR*, 2014. 2, 3

[81] Bo Wan, Desen Zhou, Yongfei Liu, Rongjie Li, and Xuming He. Pose-aware multi-level feature network for human object interaction detection. In *ICCV*, 2019. 3

[82] Eric A Wan and Rudolph Van Der Merwe. The unscented kalman filter for nonlinear estimation. In *Adaptive Systems for Signal Processing, Communications, and Control Symposium*, pages 153–158, 2000. 4

[83] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese. Densefusion: 6d object pose estimation by iterative dense fusion. In *CVPR*, 2019. 3, 4

[84] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *CVPR*, 2011. 3

[85] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 3

[86] Tiancai Wang, Rao Muhammad Anwer, Muhammad Haris Khan, Fahad Shahbaz Khan, Yanwei Pang, Ling Shao, and Jorma Laaksonen. Deep contextual attention for human-object interaction detection. In *ICCV*, 2019. 3

[87] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018. 8

[88] Shu-Fai Wong, Tae-Kyun Kim, and Roberto Cipolla. Learning motion categories using both semantic and structural information. In *CVPR*, 2007. 3

[89] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *CVPR*, 2019. 1, 3

[90] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017. 3

[91] Tete Xiao, Quanfu Fan, Dan Gutfreund, Mathew Monfort, Aude Oliva, and Bolei Zhou. Reasoning about human-object interactions through dual attention networks. In *ICCV*, 2019. 3

[92] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*, volume 32, 2018. 3, 6, 8

[93] Qi Ye, Shanxin Yuan, and Tae-Kyun Kim. Spatial attention deep net with partial pso for hierarchical hybrid hand pose estimation. In *ECCV*, 2016. 3

[94] Shanxin Yuan, Guillermo Garcia-Hernando, Björn Stenger, Gyeongsik Moon, Ju Yong Chang, Kyoung Mu Lee, Pavlo Molchanov, Jan Kautz, Sina Honari, Liuhao Ge, et al. Depth-based 3d hand pose estimation: From current achievements to future goals. In *CVPR*, 2018. 3

[95] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *ECCV*, 2018. 3

[96] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018. 2

[97] Penghao Zhou and Mingmin Chi. Relation parsing neural network for human-object interaction detection. In *ICCV*, 2019. 3

[98] Christian Zimmermann and Thomas Brox. Learning to estimate 3d hand pose from single rgb images. In *ICCV*, 2017. 3

[99] Christian Zimmermann, Duygu Ceylan, Jimei Yang, Bryan Russell, Max Argus, and Thomas Brox. Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In *ICCV*, 2019. 2, 7

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 14:25:18 UTC from IEEE Xplore. Restrictions apply.