Interaction Replica: Tracking Human–Object Interaction and Scene Changes From Human Motion

Vladimir Guzov [1,2] Julian Chibane [1,2] Riccardo Marin [1] Yannan He [1]
Yunus Saracoglu [1] Torsten Sattler [3] Gerard Pons-Moll[1,2]

[1]Tübingen AI Center, University of Tübingen, Germany, [2]Max Planck Institute for Informatics, Saarland Informatics Campus, Germany
[3]CIIRC, Czech Technical University in Prague, Czech Republic

{vladimir.guzov, riccardo.marin, yannan.he, gerard.pons-moll}@uni-tuebingen.de,
jchibane@mpi-inf.mpg.de, yunus.saracoglu@student.uni-tuebingen.de, torsten.sattler@cvut.cz

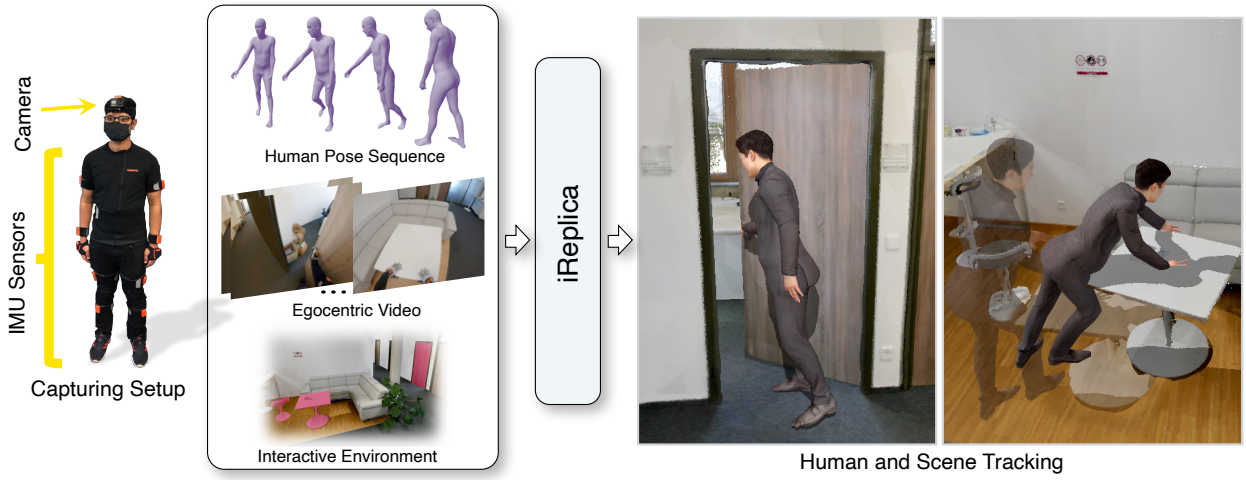

Figure 1. **Interaction Replica (iReplica)**. iReplica estimates location and full 3d pose of a subject within a large 3D scene and dynamically tracks changes made to the scene by the subject - using only wearable sensors (*left*), removing the need of external sensors. We obtain an approximate 3D human pose sequence using IMU sensors and use head camera self-localization to localize the subject in the prescanned 3d interactive environment scene. iReplica predicts human-scene contacts and updates the scene in case of interaction.

Abstract

Our world is not static and humans naturally cause changes in their environments through interactions, e.g., opening doors or moving furniture. Modeling changes caused by humans is essential for building digital twins, e.g., in the context of shared physical-virtual spaces (metaverses) and robotics. In order for widespread adoption of such emerging applications, the sensor setup used to capture the interactions needs to be inexpensive and easy-to-use for non-expert users. I.e., interactions should be captured and modeled by simple ego-centric sensors such as a combination of cameras and IMU sensors, not relying on any external cameras or object trackers. Yet, to the best of our knowledge, no work tackling the challenging problem of modeling human-scene interactions via such an ego-centric sensor setup exists. This paper closes this gap in the literature by developing a novel approach that combines visual localization of humans in the scene with contact-based reasoning about human-scene interactions from IMU data. Interestingly, we can show that even without visual observations of the interactions, human-scene contacts and interactions can be realistically predicted from human pose sequences. Our method, iReplica (Interaction Replica), is an essential first step towards the egocentric capture of human interactions and modeling of dynamic scenes, which is required for future AR/VR applications in immersive virtual universes and for training machines to behave like humans. Our code, data and model are available on our project page at http://virtualhumans.mpi-inf.mpg.de/ireplica/.

1. Introduction

Current augmented and virtual reality (AR/VR) applications show promising potential: interesting applications include collaborative developments, virtual meeting rooms, and personal assistants that help users navigate the world. While it is clear that for an immersive experience blending real and digital worlds is crucial, the current AR/VR experience is restricted to small spaces, i.e., in general, a few square meters, possibly free from objects. But consider daily actions like moving across rooms, opening and closing doors, or gathering chairs around a table. Even these

simple actions are not easy to capture with present technology, which limits the scope of AR/VR applications.

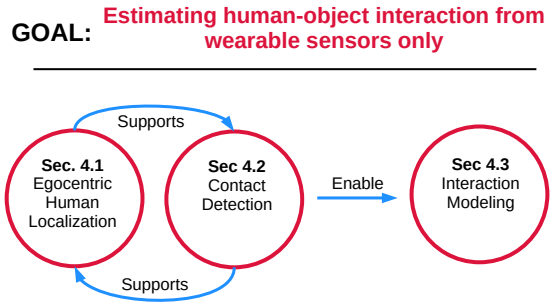

Figure 2. **Problem subdivision.** We demonstrate that joint integration of different sub-research problem improves and support each other. We show this is fundamental to achieve our goal of estimating human-scene interaction from wearable sensors only.

The predominant approach for 3D human motion estimation relies on external cameras [18, 26, 31, 32, 36, 42, 49, 80]. Yet, asking non-technical users to mount and calibrate complex multi-camera systems is clearly infeasible. Body-mounted sensors, *e.g.*, cameras and IMU sensors, seem much more ready for mass adoption.

Prior ego-centric trackers such as HPS [19], EgoLocate [75] or HSC4D [12] estimate human movement and position the person by combining head camera visual localization with IMU-based pose estimation. Methods like HPS, however, do not track scene changes. For example, if a person opens and walks through a door, such methods will only localize the person but can not infer the door movement, creating implausible reconstructions; see Figure 7, HPS.

In this work, we address, for the first time, the problem of human-scene interaction capture *from wearable sensors only*. Localizing the person with sufficient precision to track scene changes is hard, let alone estimating object motion. A major challenge is that the object is often not visible or is only partially visible in the camera; see Figure 3. In addition, since the head camera is in motion, the object's motion relative to the static world can not be directly inferred.

Since no external sensors can measure the scene changes directly, how can we predict them? Our key observations and findings are that 1) contact poses are distinctive and can be detected without visual clues, 2) knowing contact time stamps can regularize human localization, 3) objects move when the human contacts them[1]. Motivated by this, we propose *iReplica - Interaction Replica*, a novel human-centric method that automatically localizes the human in the scene (1. egocentric human visual localization), detects the time of contact and release with the object (2. contact time detection), and infers object motion based on contacts and human motion (3. interaction modeling). While works exist in each of these three sub-areas of research, no work integrates them simultaneously.

We needed several scientific innovations to integrate the aforementioned three sub-areas of research successfully. First, we improve human visual localization (HPS [19]) by optimizing the human trajectory to match reliable head camera poses and detect spatio-temporal contacts. Second, we train a transformer-based contact time detection approach based solely on the human pose, which achieves a remarkable accuracy of 0.91 and an average precision of 0.81. Third, based on the refined human visual localization in the scene and the accurate contact predictions, we infer object motion coherent with the human. Our results demonstrate that joint integration is beneficial (Fig. 2). The contact time information can be used to regularize visual localization by forcing the virtual human to contact the scene. Having precise human localization in the scene, along with contact timestamps, allows us to infer 1) where contacts occur and 2) the object's motion without seeing the object or contacts in the camera.

During this project, we captured two new datasets. To train a contact detection method, we captured a dataset of 8 subjects and more than 3 hours of human-scene interactions annotated with contact time stamps. To validate our proposed method, we captured a dataset with subjects moving and interacting with different objects in large 3D scenes. Our experiments show that iReplica can capture, for the first time, full interactions, including the human motion, its location within the 3D scene and the scene changes, all from wearable sensors alone. We demonstrate that our human-centric approach outperforms baselines, which rely on SOTA camera-based contact detection or visual object localization [13, 57].

In summary, our contributions are the following:

1. *Novel Problem & Method*: We are the first to tackle capturing human-scene interactions while localizing the human in the scene from wearable sensors alone. We propose a method to address this problem, obtaining, *for the first time*, a digital replica of the human interaction in the scene without any external cameras.
2. *Novel Data & Metrics*: We provide H-contact – a dataset of 2300+ human-scene interactions with ground truth annotated contacts. Additionally, we provide EgoHOI – a dataset of human-scene interactions in scanned environments. We propose metrics to measure the visual plausibility of reconstructed interactions and the accuracy of contact prediction and object localization.

To foster progress in this new research area, we release the method code, the evaluation protocol, and the datasets, including scans of the scenes, human motion capture aligned with them, and annotated contact timestamps.

[1]In this work we only consider static objects moved by the captured human.

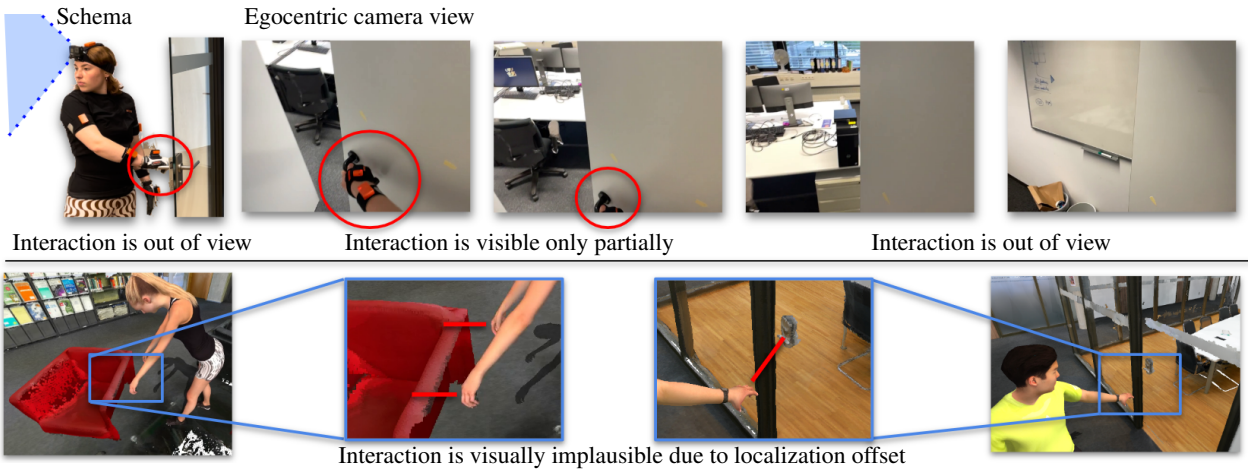

Figure 3. **Challenges.** Top row: The prediction of human-scene contacts (red circles) is hard because the interactions are frequently not in the camera view. Bottom row: Virtual replica of human pose and localization by prior work HPS [19]. HPS achieved great progress in localizing humans solely by wearable sensors (camera+IMUs). However, for our task, the localization error of 4–16 cm (red lines) leads to visually implausible results for scene interactions.

2. Related Work

Human–object interaction. Majority of current methods that record human–object interactions use external cameras. Methods to capture the full body pose also use external cameras and mostly static scenes [20, 23, 42, 80] or work with a single dynamic object without any scene context [4, 24, 62, 70, 73, 79]; similarly for human–scene and human–object interaction generation methods [21, 61, 63, 82]. A few exceptions exist though, *e.g.* RigidFusion [71] can track objects with an external RGBD sensor, and some pose estimation methods work with first-person view footage. However, those methods study the upper body or are limited to static cameras, *e.g.*, hand–object pose estimation [15, 37, 40, 46, 64, 69], or do not model dynamic objects [83]. Some methods can use egocentric video to predict the contact [13, 57], but they do not further process this information to infer object position. Our method works with body-mounted sensors and a moving camera while capturing the full-body pose and object position.

Embodied research. Body-mounted sensor setups are heavily used to solve various tasks: activity recognition methods [3, 10, 17, 45, 51, 76] use egocentric cameras looking at the body. However, they typically concentrate on capturing the upper body. Many full-body capturing methods [50, 66, 72] work with similar head-mounted setups, but as the cameras are pointed at the person wearing them rather than outwards, these methods ignore the environment around the subject. Some methods work with an outwards-facing camera [28, 77, 78]. However, they do not use any additional sensors to capture the body pose and predict it using motion priors. On the opposite side, methods like [29, 74] use inertial sensors to capture the body pose, but lack of visual cues results in an accumulating position drift, and body poses far from the ground truth. [39] proposes action recognition and localization using a first-person perspective video but does not model scene change. [81] works with a head-mounted camera, but uses it to capture the pose of other subjects, making it an external-camera method. Most related to our method, HPS [19] and following works [12, 75] capture human motion using body-mounted IMUs and a head-mounted camera looking outwards to capture the subject location within a pre-built 3D scan of the environment. We extend HPS by not only tracking the human pose but also an object the person interacts with. Whereas HPS is restricted to static environments and cannot model scene changes caused by human–object interactions, iReplica removes these restrictions.

Visual localization. Visual localization aims to estimate the pose of a camera in a known environment. Current state-of-the-art approaches are based on 2D–3D matches between pixels in the camera image and 3D scene points. These 2D–3D matches are either estimated based on local features [25, 38, 44, 53, 55, 56] or by regressing a 3D point coordinate for each pixel [5, 6, 11, 14, 59][2]. A recent line of works focuses on the robustness of localization algorithms [14, 27, 65, 68], *e.g.*, to illumination, weather, and seasonal changes, as well as to changes caused by human actions (rearranging furniture, *etc.*). These approaches assume that a large enough part of the scene remains static and is observed by the camera to facilitate pose estimation. The second assumption is violated in our scenario and we use IMU-based human pose tracking to bridge gaps where visual localization algorithms will most likely fail. As in [19], for the head-mounted camera localization, we use a state-of-the-art localization pipeline [53, 54]. Similar to the idea

[2]We refer the interested reader to [7] for a discussion and comparison of both types of approaches.

of visual-inertial approaches, *e.g.* [9, 30, 43], we use data from the IMU sensor to stabilize the predicted camera trajectory during periods of low scene visibility or when a lot a scene changes are happening. Note that our main contribution, *i.e.*, jointly reasoning about human and object pose, is not tied to any particular localization algorithm.

3. Problem Setting

Goal. We aim to *estimate human-object interaction from wearable sensors only*, without information from external sensors, using only body-mounted IMUs and an egocentric camera. This opens a broad set of interconnected challenges: how do we define the interaction? How do we detect the start and the end of it? And how do we track the object's motion without having sensors dedicated?

Assumptions. We assume a static 3D scan of the scene, along with a set of marked interactive objects, knowing their initial position and degrees of freedom (*e.g.*, a sofa can slide on the ground but cannot be lifted, or a door rotates around a hinge). We refer to this as *interactive environment* (IE).

Input/Output. We require a set of body-mounted wearable IMUs (we use 17 sensors from XSens [48]) and a video stream from a head-mounted camera. Relying only on wearable sensors lets us handle large scenes consisting of multiple rooms. Compared to external cameras, wearable sensors are much more consumer-friendly as they are easier to set up. iReplica outputs a virtual replica of the interaction, *i.e.*, coherent human and object motion in the scene.

4. iReplica

Overview. We obtain initial localization and pose estimation for the person relying on an improved version of HPS [19] (Sec 4.1, Fig 4 A). Our method considers only the human pose at each instant and predicts the probability of contact with an object (Sec 4.2, Fig 4 B). Once the contact is detected, we model the object dynamically as follows: we deform the human trajectory to match the object contact; the object is attached to the human and driven in space according to its degrees of freedom (Sec 4.3, Fig 4 C); when our method infers from the human pose the end of the contact, the object is released (Fig 4 D).

4.1. Egocentric human visual localization.

Problem. Our method is built on a combination of IMUs and head-mounted camera data. Previous methods rely on optimizations to get from these two modalities smooth trajectories estimation [19, 30, 43]. However, no previous approach considers the human's interaction with the scene nor shows extensions to incorporate constraints coming from this. Also, if 10 cm of error (average for HPS [19]) might not seem much for human localization in a building, for human-object interaction (which is our ultimate goal), this can cause dramatic inconsistencies. Instead, we see (and take advantage of) the relation between human localization and contact prediction: solving for contact prediction supports human localization in large volumes; human localization helps detect object contact in time and space.

Trajectory optimization. We start introducing an improvement over the HPS approach [19]. We deploy a simple optimization that is flexible and can be used to incorporate interaction constraints. While we work with 3D trajectories, we consider a 2D optimization since one dimension (gravity axis) is constrained by the ground of the scene. Consider the trajectory described as a 2D curve $\mathbf{l}(\tau) = (x(\tau), y(\tau))$ defined in the time interval $\tau = [\tau_{\text{start}}, \tau_{\text{end}}]$, and a list of K control points $\mathbf{p} = \{\mathbf{p}_i = (x_i, y_i)\}_{i=1}^{K}$ (constraints) at times $\tau_{\text{start}} \leq \tau_1, \ldots, \tau_K \leq \tau_{\text{end}}$. We want to recover a new trajectory $\hat{\mathbf{l}}(\tau) = (\hat{x}(\tau), \hat{y}(\tau))$ that gets close to the control points while not deviating too much from the initial trajectory. We introduce an energy E_{tr} that encodes the trajectory deviation in terms of angles.

$$E_{tr}(\hat{\mathbf{l}}, \mathbf{l}) = \int_{\tau_{\text{start}}}^{\tau_{\text{end}}} \left(\frac{d\hat{\alpha}(\tau)}{d\tau} - \frac{d\alpha(\tau)}{d\tau} \right) d\tau, \qquad (1)$$

where:

$$\hat{\alpha}(\tau) = \text{atan2}\left(\frac{d\hat{y}}{d\tau}, \frac{d\hat{x}}{d\tau}\right), \quad \alpha(\tau) = \text{atan2}\left(\frac{dy}{d\tau}, \frac{dx}{d\tau}\right).$$

Concretely, E_{tr} measures the difference between two trajectories at each instant in terms of direction (angle) variation. We define the difference only in terms of angles since, as pointed out in previous works [19], the total distance tracked by the IMUs is well measured, while the curvature tends to accumulate drift over time.

We then correct the human trajectory by optimizing the following energy:

$$F_{tr}(\mathbf{l}, \mathbf{p}) = \arg\min_{\hat{\alpha}} \left(\sum_{i=1}^{K} (||\hat{\mathbf{l}}(\tau_i) - \mathbf{p}_i||_2) + \lambda E_{tr}(\hat{\mathbf{l}}, \mathbf{l}) \right), \qquad (2)$$

where λ is the global rigidness coefficient, which encodes how much local angles should retain the initial estimation.

Contact-based human trajectory correction. In iReplica, we perform the above optimization two times. We consider the input trajectory recovered by the IMUs, and we optimize it using the control points returned by the camera localization. Then, our method detects the moments of contact along the human motion sequence. For each detection, we select the nearest object in the scene within a reasonable range (*e.g.*, 50 cm). The contact is ignored as a false positive if no object is that close. We select a contact point $\mathbf{p}_c$ as the closest point of the object to the contacting hand. Then, we rerun our optimization again, considering $\mathbf{p}_c$ as the only control point to satisfy. We report details in supplementary.

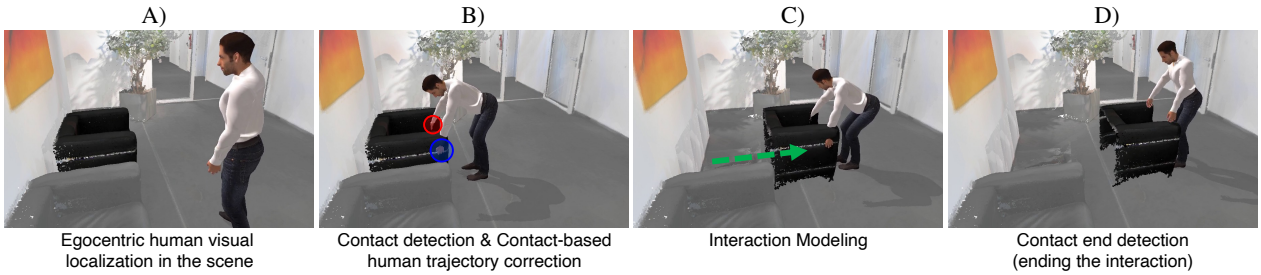

Figure 4. **Overview of iReplica.** iReplica estimates a subject's location and full pose within a large 3D scene and dynamically track changes made to the scene by the subject – using only wearable sensors. We do so in 4 steps: **A)** We obtain an initial localization of the subject in the IE by head camera self-localization. **B)** The start of the interaction is predicted by a neural network. Predictions are provided as contact / no-contact classification of the subject's hands (red and blue areas). The contacts are used to correct head camera localization of the subject, snapping the human trajectory smoothly to the object. **C)** The motion of the contacted regions is used to infer the object trajectory (green). **D)** The network predicts the release, essential to stop object dragging. The algorithm is detailed in Sec. 4.

4.2. Contact detection

Problem. The key ingredient for accurate localization is detecting when and where the user interacts with an object. In this work, we purely focus on human poses (obtained from IMUs) for contacts instead of relying on camera data for multiple reasons: (1) contacts are often not visible, *i.e.*, camera data alone is insufficient for the task. (2) IMUs are cheaper and much more power-efficient than cameras. At the same time, processing their lower-dimensional output requires significantly less compute (and thus power). This makes purely IMU-based contact prediction very attractive for applications running on mobile devices such as AR/VR headsets or robots. Naturally, combining inertial and visual data should improve performance, similar to visual-inertial localization. However, we leave this integration for future work and focus on IMU-only contact detection.

Training data. Existing datasets for human-object contact prediction contain only a limited number of samples per object type [4], or only hand-held objects [62]. In our context, the interaction involves large objects appearing in real scenes. Hence, we collect and annotate a training dataset (H-contact, Sec. 5.1) of ∼680k pose frames ($>$ 3 hours) recorded with 8 subjects wearing IMUs and 12 different objects. Our dataset is noticeably bigger compared to several other human–object and human–scene interaction datasets (BEHAVE [4] contains ∼15k frames, PROX [20] ∼100k).

Transformer. To predict contacts, we train a sequence-to-sequence Transformer [67] to map a sequence of poses to a sequence of per-hand contact probabilities. Specifically, we concatenate 61 SMPL pose vectors in a sequence, forwarding them to an MLP, appending the frame position as positional encoding, and processing them with a Transformer to output a sequence of contact probabilities for each hand. We use a sliding-window approach, and at each instance, we retain only the central (30^{th}) prediction. The contact is considered active once the probability reaches a certain threshold. The architecture is visualized in Fig. 5. To remove false negatives, any gap of $\leq 0.5\,\text{s}$ between two active contacts is filled (*i.e.* marked active). This produced the best results on a validation set (see supplementary).

While focusing on hands is not entirely descriptive of how humans interact with the world, it covers most cases in which humans cause changes in their environments. Our method can easily extend to other body parts; more detailed analyses are left for future works.

Contact intervals. Each group of consecutive frames with active contact is considered a *contact interval*. If the network predicts the end for one hand while the other is still considered to be in contact, iReplica splits the contact interval into two interactions (a two-handed and a one-handed one). Similar cases (*e.g.*, interchanging hands) are treated the same way. Likewise, our method can handle multi-object interaction – please see supplementary for details.

Training details. We train the network for 100 epochs with a batch size of 100 using the Adam optimizer [34] with a learning rate of 10^{-3} and a binary cross-entropy loss. The resulting architecture has 21.9k parameters and an inference time of less than a second per minute of motion (3600 motion windows) on an Nvidia RTX 3090 GPU.

4.3. Interaction modeling

The benefits of iReplica's pose-based contact prediction and human localization are best visualized by dynamically adapting the scene changes as their consequence. Concretely, when contact with an object is predicted, we attach the object to the user; its dynamic is driven by human motion given through IMU pose and the object's degrees of freedom defined in the interactive environment. Please see the supplementary paper for the technical details.

5. Experiments

5.1. Datasets

In this work we captured and annotated two new datasets: **H-contact** and **EgoHOI**, which we release together with our annotation tool.

H-contact is a dataset of human–object interactions, designed to serve as a training set for our contact predictor.

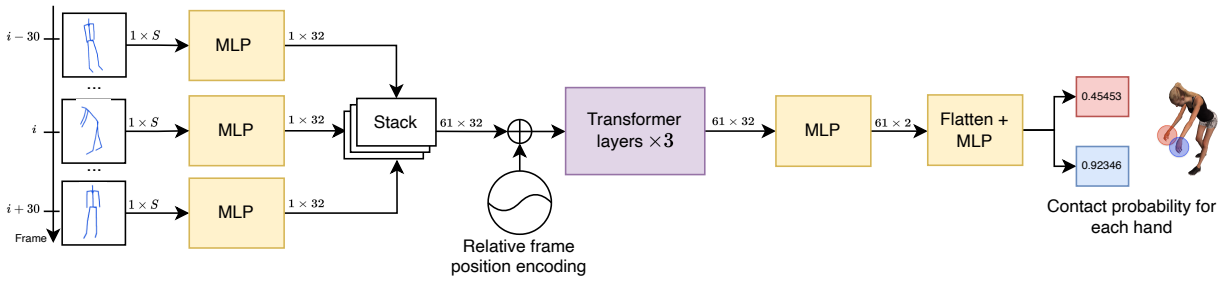

Figure 5. **Contact prediction based on human pose.** Interactions are frequently unobserved in an egocentric view, see Fig. 3 (top row), making contact prediction ill-posed. Instead, we propose to predict from sequences of full 3D human poses. We leverage a transformer-based architecture that takes 61 frames $\{i-30, \ldots, i+30\}$ of SMPL pose vectors of size $S = 69$ and predicts the contact probability for each hand for the middle frame i. See Sec. 4.2 for details.

We captured and annotated more than 2300 human–object interactions in >3 hours of recordings divided into 30 uninterrupted sequences. A total of around 680k frames, providing interaction for 8 subjects and 12 objects, whose lengths range from 1 to 19 seconds. To obtain ground-truth contact labels, we built a GUI-based annotation tool for this task. Using synchronized video from an external camera, we asked annotators to define the contact classifications.

Egocentric Human–Object Interactions (EgoHOI) is a dataset of humans performing everyday interactions with objects in real scenes recorded with wearable sensors. The sensors are placed directly on the human to allow for large recording volumes not restricted by external camera placement. The wearable setup consists of the IMU-based motion-capturing suit Xsens Awinda [48], allowing us to obtain human pose sequences, and a head-mounted RGB camera for visual localization of the subject in the scene. The dataset also includes the related interactive environments (IE): a 3D scans of the scene, segmented objects and their degrees of freedom. EgoHOI contains interactions with 14 objects (tables, windows, doors, drawers, sofas, chairs, *etc*.) in multiple IE for a total of more than 100k motion frames. We also recorded RGBD data from an external multi-camera setup to measure reconstruction accuracy.

5.2. Baselines

Due to the novelty of the proposed human–object tracking task, no published baselines exist. We introduce novel baselines and briefly describe them, see supp.mat. for details.

HPS. We compare to HPS [19] that localizes the human within the prescanned scene using the images of the head-mounted camera. HPS does not reason about human–object interactions and does not track scene changes.

HPS w/ GT combines HPS with ground-truth data to predict the object's motion. It has access to the ground-truth final object pose and ground-truth start and end time of the object interaction. To obtain the object motion estimate, it linearly interpolates the object poses in the time window. Obviously, the required ground-truth data for HPS w/ GT is not available in real-world applications. This baseline is used to show that a simple linear interpolation model is not sufficient for real-world scenes.

HPS w/ RGB Obj. Loc. localizes the object using solely the RGB frame from the head-mounted camera. As HPS uses a visual localization algorithm [53] to localize the camera in the scene, we use the same process to localize the interacted object w.r.t. the camera. Knowing the relative pose of the object to the camera localization in world space, we can estimate the object pose in world space. To simplify the matching process, this baseline uses GT object segmentations of the objects of interest.

iReplica w/ HOD [57] and iReplica w/ VISOR [13]. Our approach predicts human-scene contacts based on human pose information. Alternatively, the head-mounted RGB camera can be used to make these predictions. As baselines, we use two state-of-the-art, pretrained, RGB-based hand-contact understanding methods: **HOD** [57] and **VISOR** [13]; both predict 2D hand and object locations, and contact probabilities per hand. We use the contact probabilities as a drop-in replacement for the contact predictor in iReplica, while keeping the rest of the method fixed.

5.3. Results

Qualitative results. Fig. 6 shows our results for sample frames from multiple-scene interactions. Videos are included in the supp. mat. and we urge the reader to look at them as interactions are best judged in motion. Our results show that egocentric motion data alone can localize the human in the scene, model the interaction between the human and objects, and update the scene accordingly. Our contact predictor allows iReplica to estimate object tracking only based on the human pose; *e.g*., doors, chairs, and tables can be interacted with in the scene.

Qualitative comparisons to baselines. Fig. 7 visually compares iReplica to our baselines by showing individual frames from some of the interactions. The interactions are best viewed in the supp. video. *HPS* does not track scene changes and thus obtains unrealistic motions. For example, the door opening is not tracked. The subject should have opened the door with the handle, but the door is still closed.

Figure 6. **Qualitative results.** We show three examples of human interaction, pairing the head-mounted camera view with the interaction modeling achieved by iReplica. The object is not always visible during the interaction (Interaction 1), hand grasping can be difficult to understand from the camera (Interaction 2), or object occludes a majority of the first person view (Interaction 3). By relying on human-centric contact detection, iReplica achieves reliable modeling in all these challenging scenarios. Please see our video for more results.

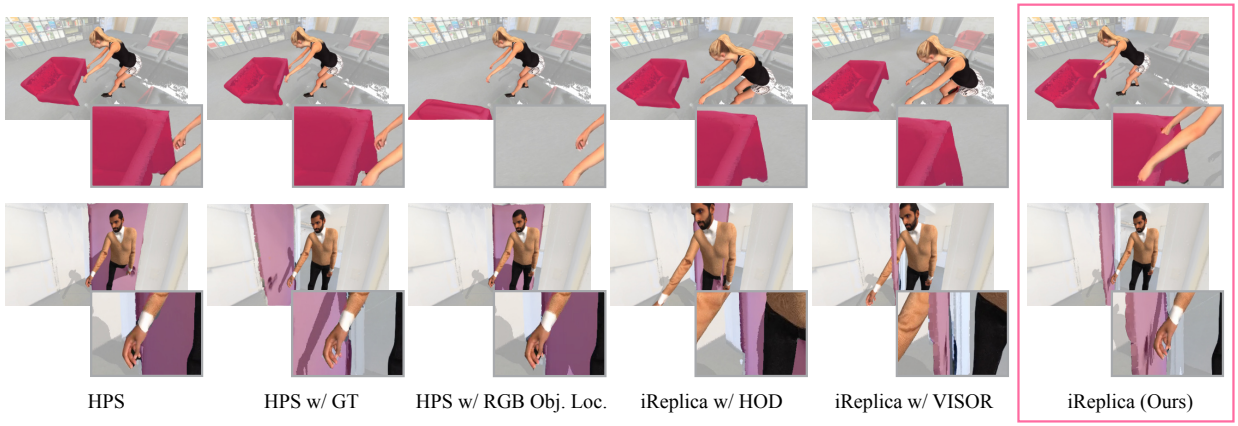

Figure 7. **Qualitative comparison.** We compare iReplica (ours) to the baseline methods (interacted object highlighted in red for visual clarity). For the sofa sequence (top row) no baseline can track the sofa and correctly place the subjects' hands. Similarly, the door (bottom row) is incorrectly placed by all baselines, and the hand is not in contact with the handle. In contrast, iReplica obtains visually plausible results by adjusting human and object locations to satisfy contact constraints.

(Fig. 7, door). Or the sofa is dragged by the subject, but the object stands still. The sofa, therefore, is visibly not in contact with the subject's hands (Fig. 7, sofa). *HPS w/ GT* linearly interpolates the object motion given ground-truth object start and end pose and contact times. The resulting interaction is not visually plausible due to a large mismatch between the subject's hands and the sofa. Human-scene interaction motion is highly non-linear, so linear approximations seem unrealistic: according to our user study, iReplica results were preferred over this baseline and HPS in 84.2% of the cases (see suppl.). *HPS w/ Obj. Loc.* fails to detect the accurate object position during the interaction, resulting in misaligned results, to the point that it fails to localize the object at all (*e.g.*, Tab. 2, Box). *iReplica w/ HOD* and *iReplica w/ VISOR* both suffer from false negatives, resulting in a sudden contact loss in the middle of the interaction and missed contacts between object and subject. *iReplica* ensures that the subject's hands are close to the object during the whole interaction – a key aspect of visual plausibility not achieved by the baselines. This shows the value of iReplica's correction of the human trajectory based on the human–object interaction.

Reconstruction quality compared to real scenes. We quantitatively validate iReplica's object and human localization results in terms of the reconstruction quality with respect to the original scene. We measure deviations from the

Error ↓	Method	Door	Sofa	Table	Box	All
Distance (in cm)	**HPS**	79.27	69.54	25.31	41.92	60.81
	HPS w/ RGB Obj. Loc.	28.66	1684.06	119.59	—	597.83
	iReplica w/ HOD [57]	57.50	55.78	1.62	**3.33**	38.58
	iReplica w/ VISOR [13]	43.40	66.74	5.98	11.31	39.59
	iReplica w/o Contact corr.	18.54	11.70	1.84	7.79	11.68
	iReplica (Ours)	**9.97**	**6.66**	**0.90**	7.09	**6.88**
Angle (in °)	**HPS**	109.19	23.53	12.16	3.76	46.89
	HPS w/ RGB Obj. Loc.	34.36	118.02	60.08	—	61.43
	iReplica w/ HOD [57]	75.74	7.74	0.78	**2.71**	28.41
	iReplica w/ VISOR [13]	56.64	17.36	2.87	12.78	27.27
	iReplica w/o Contact corr.	22.16	**5.83**	0.88	4.81	10.28
	iReplica (Ours)	**12.94**	**5.83**	**0.43**	4.81	**7.13**

Table 1. **Object localization accuracy.** Distance (in cm) and angle (in °) between object center at the end of the interaction in the GT pose and object center in the pose predicted by the algorithm.

virtual replica to the real scene using the EgoHOI dataset. Tab. 1 shows the object localization accuracy at the end of the interaction, where the GT object pose was annotated. On average, iReplica improves the results considerably (col. *All*). All object types are localized with a distance below 10 cm and an orientation error below 13 degrees.

Ablation of Contact-based human trajectory correction. We ablate the contact-based human trajectory correction by excluding it from iReplica. We report the results in Table 1 and 2 (**iReplica w/o Contact corr.**). The method greatly benefits from the proposed correction. We measure the error of human localization with (iReplica) and without (HPS [19]) correction on a special sequence that additionally has ground truth point clouds obtained via an external multi-view system of depth cameras. iReplica again improves upon HPS – see the supp. mat. for details.

Visual plausibility. To measure the visual plausibility of iReplica results compared to the baselines, we consider the contact between the human and the object. In particular, we measure the mean distance from the object to the interacting hand, see Tab. 2. iReplica keeps this distance below 3 cm. Tracking contacts and using them for attaching the object to the human motion creates the lowest distances.

Contact prediction accuracy. We benchmark the accuracy of iReplica contact prediction in isolation in Tab. 3, comparing it to our two RGB contact prediction baselines, HOD [57] and VISOR [13]. We treat the network predictions as probabilities in a binary classification task and compute 4 metrics: Average Precision (AP), Precision, Recall and Accuracy on the binarization threshold of 0.5. Our contact prediction, solely based on the 3D human pose, significantly outperforms the RGB-based reasoning - one cause is that interaction is not always visible in the camera. Once more, we remark on how 3D human poses in isolation is a highly informative indicator of interaction contacts.

6. Discussion and Conclusion

We proposed the novel problem of capturing human-scene interactions and dynamic 3D scenes solely from wearable sensors - that is, IMUs and a head-mounted camera, and not relying on any external cameras or object trackers. We show that egocentric data alone can be used to localize the human in the scene, model the interaction with the objects, and update the scene accordingly. iReplica enhances results of human localization from HPS, correcting that the human and the interacted object are close to each other.

Class label	Door	Sofa	Table	Box	All
HPS	46.00	38.32	26.35	6.64	33.61
HPS w/ GT	17.28	6.90	7.55	6.74	10.44
HPS w/ RGB Obj. Loc.	65.26	724.63	136.27	—	287.12
iReplica w/ HOD [57]	48.42	35.96	13.31	5.52	31.26
iReplica w/ VISOR [13]	33.76	51.14	13.39	**3.84**	31.17
iReplica w/o Contact corr.	18.15	9.80	6.89	5.45	11.37
iReplica (Ours)	**2.83**	**1.46**	**3.49**	5.49	**2.93**

Table 2. **Visual plausibility of human-scene interaction.** Mean distance between the object and the contacting hand (in cm) over the interaction time.

Contact predictor	AP ↑	Precision@0.5 ↑	Recall@0.5 ↑	Accuracy@0.5 ↑
HOD [57]	0.044	0.251	0.818	0.364
VISOR [13]	0.217	0.313	0.098	0.732
Ours	**0.807**	**0.786**	**0.880**	**0.905**

Table 3. **Contact prediction performance.** Metrics obtained on our test set with subjects that are not appearing in training data.

Future work and Limitations. Our simple method has some natural limitations, which point to interesting future directions. Our approach does not consider physics or collisions, which future work can investigate using simulations similar to those available in game engines. Our work relies on a prescanned IE, in which objects degrees of freedom are known a priori. Incorporating environment reconstruction (e.g., SLAM-based models) and on-the-fly segmentation (e.g., ScanNet) could remove these assumptions. Finally, our paradigm does not consider more complex interactions and object manipulations (e.g., articulated, non-rigid). This would be an exciting research direction, especially in light of the recent availability of human-object datasets [16].

Acknowledgments: Special thanks to RVH team members, and reviewers, their feedback helped improve the manuscript. The project was made possible by funding from the Carl Zeiss Foundation. This work is supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - 409792180 (Emmy Noether Programme, project: Real Virtual Humans), German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A and the Czech Science Foundation (GAČR) EXPRO (grant no. 23-07973X). Gerard Pons-Moll is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1 - Project number 390727645. Julian Chibane is a fellow of the Meta Research PhD Fellowship Program - area: AR/VR Human Understanding. Riccardo Marin has been supported by the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 101109330.

References

[1] *Autodesk FBX Software Developer Kit*, accessed August 12, 2023. https://www.autodesk.com/developer-network/platform-technologies/fbx-sdk-2020-0. 13

[2] *Microsoft Azure Kinect*, accessed August 12, 2023. https://en.wikipedia.org/wiki/Azure_Kinect. 16

[3] Bharat Lal Bhatnagar, Suriya Singh, Chetan Arora, and C.V. Jawahar. Unsupervised learning of deep feature representation for clustering egocentric actions. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 1447–1453, 2017. 3

[4] Bharat Lal Bhatnagar, Xianghui Xie, Ilya A Petrov, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Behave: Dataset and method for tracking human object interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15935–15946, 2022. 3, 5

[5] Eric Brachmann and Carsten Rother. Learning Less is More - 6D Camera Localization via 3D Surface Regression. In *CVPR*, 2018. 3

[6] Eric Brachmann and Carsten Rother. Visual camera relocalization from RGB and RGB-D images using DSAC. *arXiv:2002.12324*, 2020. 3

[7] Eric Brachmann, Martin Humenberger, Carsten Rother, and Torsten Sattler. On the limits of pseudo ground truth in visual camera re-localisation. In *ICCV*, 2021. 3

[8] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000. 13

[9] Carlos Campos, Richard Elvira, Juan J. Gómez, José M. M. Montiel, and Juan D. Tardós. ORB-SLAM3: An accurate open-source library for visual, visual-inertial and multi-map SLAM. *IEEE Transactions on Robotics*, 37(6):1874–1890, 2021. 4

[10] Congqi Cao, Yifan Zhang, Yi Wu, Hanqing Lu, and Jian Cheng. Egocentric gesture recognition using recurrent 3d convolutional neural networks with spatiotemporal transformer modules. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017. 3

[11] Tommaso Cavallari, Stuart Golodetz, Nicholas A. Lord, Julien Valentin, Victor A. Prisacariu, Luigi Di Stefano, and Philip H. S. Torr. Real-time rgb-d camera pose estimation in novel scenes using a relocalisation cascade. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2019. 3

[12] Yudi Dai, Yitai Lin, Chenglu Wen, Siqi Shen, Lan Xu, Jingyi Yu, Yuexin Ma, and Cheng Wang. Hsc4d: Human-centered 4d scene capture in large-scale indoor-outdoor space using wearable imus and lidar. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6792–6802, 2022. 2, 3

[13] Ahmad Darkhalil, Dandan Shan, Bin Zhu, Jian Ma, Amlan Kar, Richard Higgins, Sanja Fidler, David Fouhey, and Dima Damen. Epic-kitchens visor benchmark: Video segmentations and object relations. In *Proceedings of the Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks*, 2022. 2, 3, 6, 8, 16

[14] Siyan Dong, Qingnan Fan, He Wang, Ji Shi, Li Yi, Thomas Funkhouser, Baoquan Chen, and Leonidas J Guibas. Robust neural routing through space partitions for camera relocalization in dynamic indoor environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8544–8554, 2021. 3

[15] Bardia Doosti, Shujon Naha, Majid Mirbagheri, and David J Crandall. Hope-net: A graph-based model for hand-object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6608–6617, 2020. 3

[16] Zicong Fan, Omid Taheri, Dimitrios Tzionas, Muhammed Kocabas, Manuel Kaufmann, Michael J. Black, and Otmar Hilliges. ARCTIC: A dataset for dexterous bimanual hand-object manipulation. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 8

[17] Alireza Fathi, Ali Farhadi, and James M Rehg. Understanding egocentric activities. In *2011 international conference on computer vision*, pages 407–414. IEEE, 2011. 3

[18] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7297–7306, 2018. 2

[19] Vladimir Guzov, Aymen Mir, Torsten Sattler, and Gerard Pons-Moll. Human poseitioning system (hps): 3d human pose estimation and self-localization in large scenes from body-mounted sensors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4318–4329, 2021. 2, 3, 4, 6, 8, 16, 17

[20] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J. Black. Resolving 3D human pose ambiguities with 3D scene constraints. In *International Conference on Computer Vision*, pages 2282–2292, 2019. 3, 5

[21] Mohamed Hassan, Duygu Ceylan, Ruben Villegas, Jun Saito, Jimei Yang, Yi Zhou, and Michael J Black. Stochastic scene-aware motion prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11374–11384, 2021. 3

[22] Mohamed Hassan, Partha Ghosh, Joachim Tesch, Dimitrios Tzionas, and Michael J. Black. Populating 3D scenes by learning human-scene interaction. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021. 16

[23] 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 of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13274–13285, 2022. 3

[24] Yinghao Huang, Omid Taheri, Michael J Black, and Dimitrios Tzionas. Intercap: Joint markerless 3d tracking of humans and objects in interaction. In *Pattern Recognition: 44th DAGM German Conference, DAGM GCPR 2022, Konstanz, Germany, September 27–30, 2022, Proceedings*, pages 281–299. Springer, 2022. 3

[25] A. Irschara, C. Zach, J.-M. Frahm, and H. Bischof. From Structure-from-Motion Point Clouds to Fast Location Recognition. In *CVPR*, 2009. 3

[26] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. Learnable triangulation of human pose. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7718–7727, 2019. 2

[27] Ara Jafarzadeh, Manuel López Antequera, Pau Gargallo, Yubin Kuang, Carl Toft, Fredrik Kahl, and Torsten Sattler. Crowddriven: A new challenging dataset for outdoor visual localization. In *ICCV*, 2021. 3

[28] Hao Jiang and Kristen Grauman. Seeing invisible poses: Estimating 3d body pose from egocentric video. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3501–3509. IEEE, 2017. 3

[29] Yifeng Jiang, Yuting Ye, Deepak Gopinath, Jungdam Won, Alexander W Winkler, and C Karen Liu. Transformer inertial poser: Real-time human motion reconstruction from sparse imus with simultaneous terrain generation. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 3

[30] Eagle S. Jones and Stefano Soatto. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *The International Journal of Robotics Research*, 30(4):407–430, 2011. 4

[31] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 8320–8329, 2018. 2

[32] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2018. 2

[33] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 15

[34] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 5

[35] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9799–9808, 2020. 16

[36] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5253–5263, 2020. 2

[37] Taein Kwon, Bugra Tekin, Jan Stuhmer, Federica Bogo, and Marc Pollefeys. H2o: Two hands manipulating objects for first person interaction recognition. *arXiv preprint arXiv:2104.11181*, 2021. 3

[38] Yunpeng Li, Noag Snavely, Dan P. Huttenlocher, and Pascal Fua. Worldwide Pose Estimation Using 3D Point Clouds. In *ECCV*, 2012. 3

[39] Miao Liu, Lingni Ma, Kiran Somasundaram, Yin Li, Kristen Grauman, James M Rehg, and Chao Li. Egocentric activity recognition and localization on a 3d map. In *European Conference on Computer Vision*, pages 621–638. Springer, 2022. 3

[40] Shaowei Liu, Hanwen Jiang, Jiarui Xu, Sifei Liu, and Xiaolong Wang. Semi-supervised 3d hand-object poses estimation with interactions in time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14687–14697, 2021. 3

[41] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics*, 2015. 13

[42] Zhengyi Luo, Shun Iwase, Ye Yuan, and Kris Kitani. Embodied scene-aware human pose estimation. *arXiv preprint arXiv:2206.09106*, 2022. 2, 3

[43] Simon Lynen, Torsten Sattler, Michael Bosse, Joel A Hesch, Marc Pollefeys, and Roland Siegwart. Get out of my lab: Large-scale, real-time visual-inertial localization. In *Robotics: Science and Systems*, page 1, 2015. 4

[44] Simon Lynen, Bernhard Zeisl, Dror Aiger, Michael Bosse, Joel Hesch, Marc Pollefeys, Roland Siegwart, and Torsten Sattler. Large-scale, real-time visual–inertial localization revisited. *The International Journal of Robotics Research*, 39 (9):1061–1084, 2020. 3

[45] Minghuang Ma, Haoqi Fan, and Kris M. Kitani. Going deeper into first-person activity recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1894–1903, 2016. 3

[46] Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. Generalized feedback loop for joint hand-object pose estimation. *IEEE transactions on pattern analysis and machine intelligence*, 42(8):1898–1912, 2019. 3

[47] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 15

[48] Monique Paulich, Martin Schepers, Nina Rudigkeit, and G. Bellusci. *Xsens MTw Awinda: Miniature Wireless Inertial-Magnetic Motion Tracker for Highly Accurate 3D Kinematic Applications*, 2018. 4, 6, 13

[49] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3D human pose and shape from a single color image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2

[50] Helge Rhodin, Christian Richardt, Dan Casas, Eldar Insafutdinov, Mohammad Shafiei, Hans-Peter Seidel, Bernt Schiele, and Christian Theobalt. Egocap: egocentric marker-less motion capture with two fisheye cameras. *ACM Transactions on Graphics (TOG)*, 35(6):162, 2016. 3

[51] Grégory Rogez, James S Supancic, and Deva Ramanan. First-person pose recognition using egocentric workspaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4325–4333, 2015. 3

[52] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bod-

ies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), 2017. 14

[53] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12716–12725, 2019. 3, 6, 13, 16

[54] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning Feature Matching with Graph Neural Networks. In *CVPR*, 2020. 3

[55] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & Effective Prioritized Matching for Large-Scale Image-Based Localization. *PAMI*, 2017. 3

[56] J. L. Schönberger, M. Pollefeys, A. Geiger, and T. Sattler. Semantic Visual Localization. In *CVPR*, 2018. 3

[57] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F Fouhey. Understanding human hands in contact at internet scale. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9869–9878, 2020. 2, 3, 6, 8, 16

[58] Ken Shoemake. Animating rotation with quaternion curves. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pages 245–254, 1985. 15

[59] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 3

[60] Konstantin Sofiiuk, Ilia Petrov, Olga Barinova, and Anton Konushin. f-brs: Rethinking backpropagating refinement for interactive segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8623–8632, 2020. 16

[61] Sebastian Starke, He Zhang, Taku Komura, and Jun Saito. Neural state machine for character-scene interactions. *ACM Trans. Graph.*, 38(6):209–1, 2019. 3

[62] Omid Taheri, Nima Ghorbani, Michael J Black, and Dimitrios Tzionas. Grab: A dataset of whole-body human grasping of objects. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 581–600. Springer, 2020. 3, 5

[63] Omid Taheri, Vasileios Choutas, Michael J Black, and Dimitrios Tzionas. Goal: Generating 4d whole-body motion for hand-object grasping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13263–13273, 2022. 3

[64] Bugra Tekin, Federica Bogo, and Marc Pollefeys. H+ o: Unified egocentric recognition of 3d hand-object poses and interactions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4511–4520, 2019. 3

[65] Carl Toft, Will Maddern, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, Fredrik Kahl, and Torsten Sattler. Long-Term Visual Localization Revisited. *TPAMI*, pages 1–1, 2020. 3

[66] Denis Tome, Thiemo Alldeick, Patrick Peluse, Gerard Pons-Moll, Lourdes Agapito, Hernan Badino, and Fernando de la Torre. Selfpose: 3d egocentric pose estimation from a headset mounted camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 3

[67] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 5

[68] Johanna Wald, Torsten Sattler, Stuart Golodetz, Tommaso Cavallari, and Federico Tombari. Beyond controlled environments: 3d camera re-localization in changing indoor scenes. In *European Conference on Computer Vision*, pages 467–487. Springer, 2020. 3

[69] He Wang, Sören Pirk, Ersin Yumer, Vladimir G Kim, Ozan Sener, Srinath Sridhar, and Leonidas J Guibas. Learning a generative model for multi-step human-object interactions from videos. In *Computer Graphics Forum*, pages 367–378. Wiley Online Library, 2019. 3

[70] Zhenzhen Weng and Serena Yeung. Holistic 3d human and scene mesh estimation from single view images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 334–343, 2021. 3

[71] Yu-Shiang Wong, Changjian Li, Matthias Niessner, and Niloy J. Mitra. Rigidfusion: Rgb-d scene reconstruction with rigidly-moving objects. *Computer Graphics Forum*, 40(2), 2021. 3

[72] Weipeng Xu, Avishek Chatterjee, Michael Zollhoefer, Helge Rhodin, Pascal Fua, Hans-Peter Seidel, and Christian Theobalt. Mo^2Cap^2 : Real-time mobile 3d motion capture with a cap-mounted fisheye camera. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–1, 2019. 3

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

[74] Xinyu Yi, Yuxiao Zhou, and Feng Xu. Transpose: real-time 3d human translation and pose estimation with six inertial sensors. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021. 3

[75] Xinyu Yi, Yuxiao Zhou, Marc Habermann, Vladislav Golyanik, Shaohua Pan, Christian Theobalt, and Feng Xu. Egolocate: Real-time motion capture, localization, and mapping with sparse body-mounted sensors. *ACM Transactions on Graphics (TOG)*, 42(4), 2023. 2, 3

[76] H. Yonemoto, K. Murasaki, T. Osawa, K. Sudo, J. Shimamura, and Y. Taniguchi. Egocentric articulated pose tracking for action recognition. In *International Conference on Machine Vision Applications (MVA)*, 2015. 3

[77] Ye Yuan and Kris Kitani. 3d ego-pose estimation via imitation learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 735–750, 2018. 3

[78] Ye Yuan and Kris Kitani. Ego-pose estimation and forecasting as real-time pd control. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019. 3

[79] 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 *European Conference on Computer Vision (ECCV)*, 2020. 3

[80] Siwei Zhang, Yan Zhang, Federica Bogo, Marc Pollefeys, and Siyu Tang. Learning motion priors for 4d human body capture in 3d scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11343–11353, 2021. 2, 3

[81] Siwei Zhang, Qianli Ma, Yan Zhang, Zhiyin Qian, Taein Kwon, Marc Pollefeys, Federica Bogo, and Siyu Tang. Egobody: Human body shape and motion of interacting people from head-mounted devices. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VI*, pages 180–200. Springer, 2022. 3

[82] Xiaohan Zhang, Bharat Lal Bhatnagar, Sebastian Starke, Vladimir Guzov, and Gerard Pons-Moll. Couch: Towards controllable human-chair interactions. In *European Conference on Computer Vision (ECCV)*. Springer, 2022. 3

[83] Yang Zheng, Yanchao Yang, Kaichun Mo, Jiaman Li, Tao Yu, Yebin Liu, C Karen Liu, and Leonidas J Guibas. Gimo: Gaze-informed human motion prediction in context. In *European Conference on Computer Vision*, pages 676–694. Springer, 2022. 3

SUPPLEMENTARY MATERIALS
Interaction Replica: Tracking human–object interaction and scene changes from human motion

Abstract

In this document, we report additional details and results, which we cannot include in the main manuscript for space limits: we present a general description of HPS summarizing its key aspects relevant to our method (Sec. 7); we report the details of our capturing setup (Sec. 8) and how we can incorporate data coming from motion capture gloves (Sec. 9); we detail our framework for Interaction Modeling: how we validate our false-positive pruning, model objects motion driven by human contact, and our strategy to deal with multiple objects (Sec. 10); we show an intuition behind the trajectory correction and the regularization of the penalty for angles deviations (Sec. 11); we provide information about implementation and performance (Sec. 12) and detailed descriptions of baselines (Sec. 13); additional comparison for the contact-based prediction model (Sec. 14); an external RGBD camera evaluation which further prove the relevance of human-object interaction also for human localization (Sec. 15); examples of our annotation tool (Sec. 16) and of the dataset (Sec. 17) and iReplica's features (Sec. 18). We hope that this extensive collection of resources can help the reader's intuition and ease the work on this complex problem for the future researchers.

7. Human POSEitioning System (HPS)

For completeness, we briefly introduce HPS, the system we use in our pipeline to provide initial human poses.

The HPS pipeline produces poses for the SMPL [41] model together with the global position within the 3D scan coordinate system. It uses data from the body-mounted sensors, the video from the head-mounted camera, and the 3D point cloud of the pre-scanned scene as input. The output represents the human localized in the 3D scene. HPS relies on three steps:

1) Pose estimation from XSens. From the IMU measurements, XSens uses a proprietary algorithm based on the Kalman filter and a kinematic body model, which takes physical limitations of the body into account. The system provides the human pose and translation estimation in its own coordinate frame. While these estimates can be considered accurate, the system has two main drawbacks: it accumulates errors over time, leading to significant drift, and the human dynamic is not registered within the global 3D scene, causing potential inconsistencies (*e.g.*, wall and floor penetration).

2) Head-mounted camera localization. For every camera frame, HPS provides the initial head pose estimates using a hierarchical localization algorithm [53], which operates on RGB images. Given a dataset of images with known positions in the pre-scanned scene, the algorithm establishes correspondences between those images and the camera frame, finding the position of the head camera by minimizing the reprojection loss. The camera lets HPS globally localize the human in the scene, but the raw localizations are noisy and do not ensure realistic human placement in the scene.

3) Combining IMUs and the head-mounted camera. HPS combines the two previous steps together with the 3D point cloud of the pre-scanned scene to optimize the human pose and satisfy all the previously mentioned constraints. The simultaneous optimization for the IMU-based body poses and camera localizations eliminates the noise of the camera pose predictions, and significantly decreases IMU drift at the same time, resulting in stable and accurate results.

8. Capturing Setup

We use a combination of the Xsens Awinda [48] IMU system to capture body motions and a head-mounted camera (Apple iPhone 12 or GoPro Hero 8) to capture the first-person view. The camera captures at a resolution of 1920×1440, 30 FPS. While both cameras feature automatic stabilization, we explicitly turn it off to get a better idea of the head motion. We use OpenCV [8] to calibrate the camera intrinsics. The Xsens Awinda consists of 17 IMU sensors attached to the body with velcro straps and captures the body motion at 60 FPS.

The Xsens Awinda outputs human poses in the form of skeleton joint angles, which is not directly compatible with the SMPL [41] pose parameters format. To convert between these formats, we developed the following retargeting algorithm: we export the motion from the Xsens internal format (MVN) into the Autodesk FBX format and use the Autodesk FBX Python SDK [1] to extract the rotation of each

joint as a quaternion. We convert those quaternions to the axis-angle representation used in SMPL. We map joint rotations from the FBX skeleton to the SMPL skeleton according to a manually designed mapping. Finally, we produce the SMPL pose parameter vector by concatenating the mapped axis-angle joint rotations in the right order. If available, we additionally use Xsens gloves from Manus to capture the fine-grained fingers motion, and adapt the algorithm accordingly (see Sec. 9).

9. Adding Hands Data

If fine-grained hand positions are available (*e.g.*, captured with motion capture gloves), we can modify the algorithm to consider such data. Namely, we replace the SMPL model with SMPL+H [52], which has the same template body mesh, but provides additional 30 joints (3 joints for each finger) for detailed hand pose representation. We additionally change the input of our contact prediction network to accept vectors of concatenated body pose and hand pose parameters, therefore the input becomes 61 vectors of size $1 \times \hat{S}$, $\hat{S} = 159$, adding 90 parameters of hands pose to each input vector. No additional architecture changes are made.

10. Interaction Modeling - Details

In this section, we provide more details on the object motion driven by human contact, also depicted in Fig. 8.

Processing the contact intervals. To remove false negatives, we post-process the predicted sequence and fill in gaps between active contacts that are shorter than $0.5\,\text{s}$, which produced the best results on a validation set (see the inset table).

t	Error (cm)
0.00	13.61
0.25	8.49
0.50	**7.13**
1.00	8.86
2.00	8.86

Degrees of freedom. We model the degrees of freedom (DOF) of the motion of each object as to avoid unrealistic motions. For example, a door can only rotate around a specific axis, and a sofa can only slide along the floor. While our model operates in a 3D scene, we present all derivations in 2D, since all the processed motions happen along the floor and any change in the direction orthogonal to the movement plane does not affect the object trajectory and will be removed.

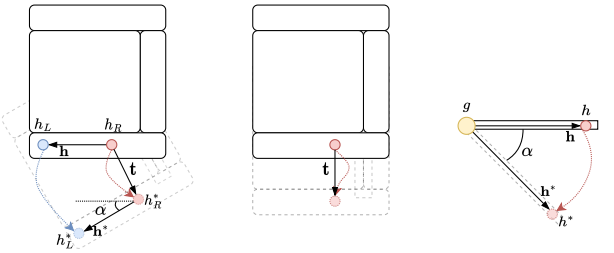

Figure 8. **Obtaining object trajectory from hand interactions**. Colored dashed lines denote hands trajectories, α is inferred rotation angle, $\mathbf{t}$ is inferred object translation vector

A) Two-hands interaction with sliding objects. We consider the point cloud of an interactive object $O \in \mathbb{R}^{n \times 3}$ that can be freely moved on a two-dimensional plane. When the two hands contact the object, we denote the position of two keypoints corresponding to the middle of each hand as $h_L = (x_L, y_L)$ and $h_R = (x_R, y_R)$ for the left and the right hand, respectively, and $\mathbf{h}$ as the vector that connects h_R to h_L. When the human moves, we register the new positions of the hands as h_L^* and h_R^*, together with the new connecting vector $\mathbf{h}^*$. Then, we compare the hand configurations to recover a translation and a rotation. Firstly, we define the translation t as

$$\mathbf{t} = (x_R^* - x_R, y_R^* - y_R). \tag{3}$$

Then, we compute the rotation angle as

$$\alpha = \arccos\left(\frac{\mathbf{h}\mathbf{h}^*}{|\mathbf{h}||\mathbf{h}^*|}\right). \tag{4}$$

Finally, we recover the 2D rotation matrix $\mathbf{R}^\alpha$ associated with the angle. The sign of α encodes the direction of the rotation, and it can be obtained by taking the cross-product between the vectors $\mathbf{h}$ and $\mathbf{h}^*$.

B) One-hand interaction with sliding objects. When only one hand h interacts with the object, without any further information about the object's physics (*e.g.*, its friction with the ground), it is impossible to recover the object rotation. Hence, given a new configuration h^*, we just compute the translation

$$\mathbf{t} = (\widehat{x} - x, \widehat{y} - y). \tag{5}$$

C) Interaction with hinged objects. In this case, the object has a hinge positioned in $g = (x_g, y_g)$. When the contact begins at the point $h = (x_h, y_h)$, we compute the vector $\mathbf{h}$ that connects g to h:

$$\mathbf{h} = (x_h - x_g, y_h - y_g). \tag{6}$$

When the human moves, we register the new position of the contact point h^* and accordingly recompute the connecting vector $\mathbf{h}^*$. Then we compute the angle α between $\mathbf{h}$ and $\mathbf{h}^*$ as in Equation 4, and we recover the associated rotation matrix $\mathbf{R}^\alpha$.

After obtaining these transformations, we apply them to the first two coordinates of each point of the object O.

$$O^* = \mathbf{R}^\alpha O + t \tag{7}$$

10.1. Multiple Objects Interaction

The contact strategy described in Sec. 4.2 of the main paper naturally extends to multiple objects interaction. When the start of contact is identified, iReplica considers the closest object within a 0.5 m radius (if any) and initiates the interaction, which lasts for the whole contact interval. After the end of the contact, the object is released, and the method waits for the next contact interval to proceed with the next interaction.

To better distinguish between the objects and improve our robustness to false positive interactions, the contact predictor of iReplica comprises a set of transformers, one for each interaction class: one for the sliding objects and one for the hinged ones. These two networks work in parallel: when one detects a starting contact, the object search in the neighbourhood is performed only for the specific category. The contact is ignored if no object of that class is identified in the user's proximity.

11. Trajectory Optimization - Details

In Equation 1 of the main manuscript, we introduced the E_{tr} energy, which limits the deviation of the trajectory, penalizing changes in the angles. Such energy is weighted by a coefficient λ in eq. 2 of the main manuscript, which defines a global rigid coefficient of the trajectory: a lower value lets the trajectory freely bent to fit the control points, while a higher value preserves the local curvature and promotes global rigidness of the trajectory. To help the reader's intuition, we report in Fig. 9 an example for different values of λ.

For completeness, we also report the definition of atan2 function:

$$\text{atan2}(y,x) = \begin{cases} \arctan(\frac{y}{x}) & \text{if } x > 0, \\ \arctan(\frac{y}{x}) + \pi & \text{if } x < 0 \text{ and } y \geq 0, \\ \arctan(\frac{y}{x}) - \pi & \text{if } x < 0 \text{ and } y < 0, \\ +\frac{\pi}{2} & \text{if } x = 0 \text{ and } y > 0, \\ -\frac{\pi}{2} & \text{if } x = 0 \text{ and } y < 0, \\ \text{undefined} & \text{if } x = 0 \text{ and } y = 0. \end{cases}$$

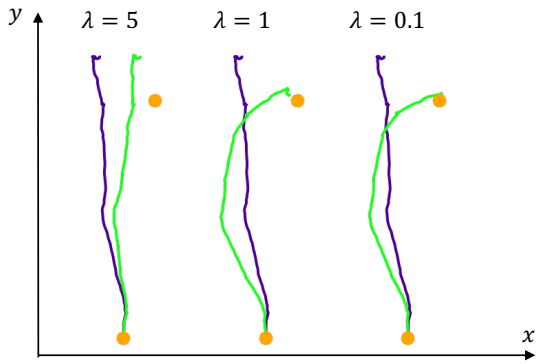

Figure 9. **Trajectory fitting with bending energy.** Bending trajectories with F_{tr} using different rigidness coefficients λ, purple marks the original trajectory, green marks the result, orange dots denote control points.

12. Implementation and Performance

We implement our algorithms in Python using the PyTorch [47] library for the contact prediction network and the bending energy optimization algorithm. For the latter, we use the Adam [33] optimizer with 1000 iterations, with the learning rate and the rigidness coefficient λ acting as hyperparameters.

The most computationally expensive part of our algorithm is the visual localization pipeline needed for HPS, requiring 8 seconds per frame on an NVIDIA Q8000 GPU, while the other steps take little additional time. Such a computationally expensive pipeline provides good localization results and supports our exploration. Note that the performance of the visual localization network itself is not the focus of our study, and we expect that more efficient alternatives can replace this algorithm in the future.

13. Baselines

In this section, we provide additional details for some of the baselines used in the paper.

13.1. HPS w/ GT

In this baseline, we assume that an oracle provides the ground-truth final object position, as well as the time window of the interaction. We stress that neither of these is available at inference time in our setting, and our method does not rely on them. Then the human motion is solely estimated using HPS, and the object movement is modeled using linear interpolation for translation and spherical linear interpolation (Slerp [58]) for rotation. By inspecting the qualitative results (please refer to the attached video), we see that motion between humans and objects happens asynchronously and, therefore, unrealistically. This baseline shows that, even in the presence of further assumptions, modelling the object trajectory is non-trivial. It is clear that human motion is rarely linear, and more sophisticated tech-

niques are required.

User study. To measure the realism of the motion produced by linear interpolation, we did a user study and asked 73 respondents to rank the interactions produced by iReplica, HPS w/ GT and HPS by realism. Each participant was given 9 questions, each showing results generated by two methods side-by-side in a random sequence. iReplica results were preferred in 84.2% of the cases, proving that our body-driven object tracking produces more realistic interaction.

13.2. HPS w/ RGB Obj. Loc

In this baseline, we assume that for each frame of the head-mounted camera, we have a perfect 2D segmentation mask for the object, obtained by semi-automatic annotation using an interactive segmentation pipeline [60]. Then we use the same localization method [53] as HPS to provide a 6-DoF localization of the object w.r.t. the human. Starting from the dataset of images with the known object positions in the pre-scanned scene, the algorithm establishes correspondences between those images and frames from the head-mounted camera. The head-mounted camera is then localized by minimizing reprojection loss. Next, the object's location relative to the camera is recovered by matching and optimizing with only the 2D key points inside the object mask. This information is combined with the camera position to recover the object's location in world coordinates. This baseline highlights that the camera is unreliable for localizing the object in the space. Detecting local landmarks is dramatically harmed by occlusions, head shaking, and the object missing in several frames.

13.3. iReplica w/ HOD

Given the availability of a head-mounted camera, we explore the possibility of using it to predict contacts. In this baseline, we replace our pose-based contact predictor with HOD [57], a method trained on many YouTube videos. Starting from a single RGB image, it predicts a full set of hand interaction properties: hands bounding box, object bounding box, and the contact state for each hand. We keep the rest of our method fixed except for the contact prediction part.

The original paper [57] shows several results from an egocentric perspective, but it also mentions failure cases when hands and objects are close to each other. We confirm this by qualitative inspection, noting that the method loses contact with the object during the interaction.

13.4. iReplica w/ VISOR

Given that HOD is trained on a variety of videos, which also include extrinsic views, we deploy a similar baseline but rely on an RGB method specifically trained on egocentric views. Specifically, we consider the baseline trained for the HOS challenge[3] on the VISOR [13] dataset. This baseline relies on PointRend [35], augmented with auxiliary detection heads to predict the contact, following the same schema of HOD [57]. As in the previous baseline, we replace our pose-based contact prediction, leaving other parts of the method untouched. However, also in this case, we observe missing contacts and wrong release prediction, often causing human penetration (*e.g.*, crossing the door).

Contact predictor	AP ↑	Precision@0.5 ↑	Recall@0.5 ↑	Accuracy@0.5 ↑
POSA [22]	0.033	0.115	0.716	0.297
Ours	**0.807**	**0.786**	**0.880**	**0.905**

Table 4. **Additional contact prediction comparison.** Additional comparison with the prediction model from POSA [22]. Model was finetuned on our training data and tested on the same testing set as in Table 3 of the main paper.

14. Contact Prediction – More Results

We present an additional comparison with a method of contact detection appearing in POSA [22]. POSA presents a human-scene interaction model that can be used as a prior for human placement in the scene. Compared to iReplica, POSA has a different goal and applications: it does not consider dynamic interactions, works only with one human pose frame at a time and is focused on static scenes. However, this is the closest baseline for our task of temporal pose-based interaction prediction. Results are presented in Table 4. For a fairer comparison, we finetune the POSA model on the same training subset from the H-Contact dataset used for iReplica; we also average contact prediction scores from a selected region of each hand and treat this as a per-hand contact probability.

The comparison shows that the POSA model could not estimate contacts reliably, even after finetuning it on our training data. One possible reason for this could be that this method works with single frames, which leads to prediction uncertainty for many poses, while iReplica's transformer-based model considers a one-second window of motion.

15. External Camera Evaluation

To additionally measure the human-object localization accuracy of our method, we recorded a special sequence with ground-truth data obtained via an external multi-view system of 3 depth cameras. The experimental setup closely follows the one used in HPS [19], however we capture the full dynamic object interaction while in HPS only the human body motion and the static scene was captured.

We use 3 calibrated Azure Kinect [2] RGBD sensors.

[3] https://github.com/epic-kitchens/VISOR-HOS

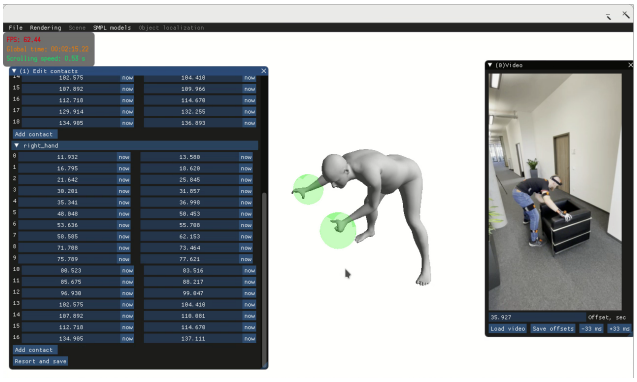

Figure 10. **Annotation tool.** The user views the RGB video frames (*right*) and annotates the start and the end of contact interaction (*left*). To help with disambiguating occlusions, the tool also shows the 3D pose (*center*) together with the annotated presence of contact for each hand (green circles).

	Human to GT $E_{body} \downarrow$	Object to GT $E_{obj} \downarrow$
HPS	9.771	22.651
iReplica (ours)	**8.981**	**8.471**

Table 5. **Human and object tracking quality w.r.t. the real scene:** Mean 3D error (in cm) between tracked and moving human/object models compared to the real scene. The scene is captured via a synchronized, multi-view RGBD video recording setup observing the interaction.

By combining the outputs of these sensors, we obtain a sequence of 3D point clouds of the scene and a subject. Each sensor outputs the depth map with a resolution of 640×576 pixels and color frames with a resolution of 2048×1536 at around 30 FPS. The Azure Kinect features built-in temporal synchronization, but to merge the output of the sensor into the scene ground truth representation, we also need to calibrate them spatially. For that we use a three-stage localization pipeline, similar to [19]: **1)** we record a special sequence with 300 frames depicting the empty scene. We localize the RGB camera in each of these frames using the same algorithm as used for the head-mounted camera and average the 300 localization results; **2)** we perform ICP between the scene 3D scan and the point cloud unprojected from the depth map; **3)** we apply manual corrections if needed. Using the obtained positions of the sensors, the point cloud representation of the scene is formed by unprojecting depth maps from all 3 sensors to 3D. To perform the evaluation, we manually synchronize the time between the iReplica motion sequence and the aforementioned point cloud representation. For each frame of the test sequence, we separately measure object and human localization accuracy E_{obj} and E_{body}:

- E_{obj}: mean Chamfer distance from the object point cloud to the ground-truth point cloud,
- E_{body}: mean Chamfer distance from the human body SMPL mesh to ground-truth point cloud.

Results are presented in Table 5. Since HPS models only humans, the large error from the object ground truth is not surprising. Although HPS and iReplica share the same camera localization principle, we observe that our iReplica improves human localization. This validates that using the detected human–object interaction to adapt the human localization trajectory helps to improve reconstruction correctness. Moreover, it drastically enhances visual plausibility, as explained in the qualitative analysis.

16. Annotation Tool

Fig. 10 is a screenshot of our annotation tool for preparing our H-contact dataset. Given a video frame (*right*) and the reconstruction (*center*), the user can annotate contacts by clicking on the interacting hands. The annotator can seek forwards and backward in the video, and the 3D reconstruction helps to disambiguate occluded poses. On average, it takes around 2–4 seconds to annotate 1 second of video.

17. Examples from the Datasets

In Fig. 11 we report some examples from the H-contact and EgoHOI datasets. For H-contact, we provide IMU measurements, SMPL parameters, contact annotations, and external camera recordings. For EgoHOI, we provide interactive scenes, recordings from head-mounted RGB cameras, IMU data, as well as contact labels and GT final object positions (for evaluation purposes).

18. Features of iReplica

Fig. 12 highlight the features of iReplica on the sequence, recorded to simulate a realistic and natural scenario. iReplica does not assume a single object interaction: instead, it applies to interactive scenes where many objects can be differently re-arranged by the interaction (*e.g.*, table, chair). iReplica also works with objects with non-linear motion in space, for example, the doors that might start and end their movement in the same place. In this case, for example, an interpolation baseline would not provide any object dynamic since the initial and final configurations are the same. Instead, iReplica tracks the full trajectory of the object. iReplica allows the user to move in the space freely and interact naturally as in everyday life. We refer to the supplementary video for the animated results.

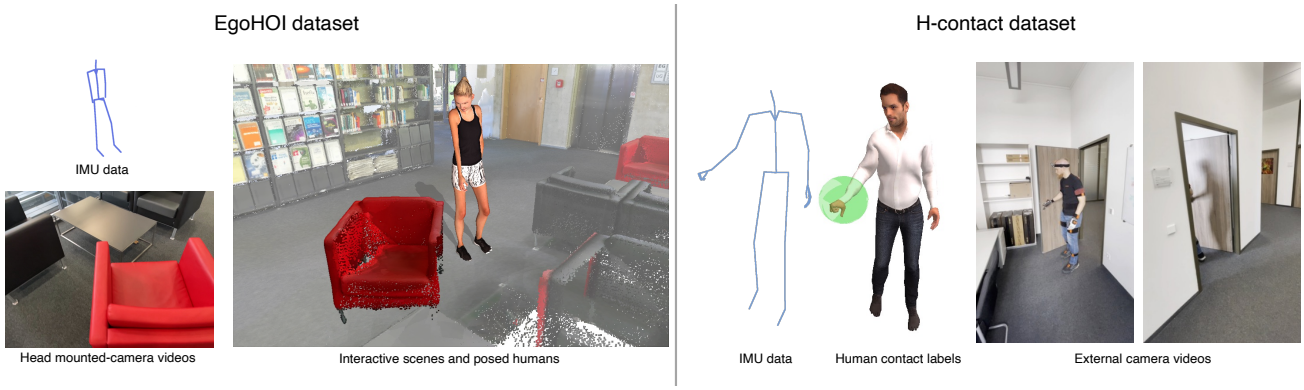

Figure 11. **EgoHOI and H-Contact examples.** For EgoHOI, we report for each timestamp the data obtained by the IMUs, the head-mounted camera frame, and the 3D posed human inside the interactive scene. For H-Contact, we provide IMUs data, contact labels for each hand, and recordings from external cameras.

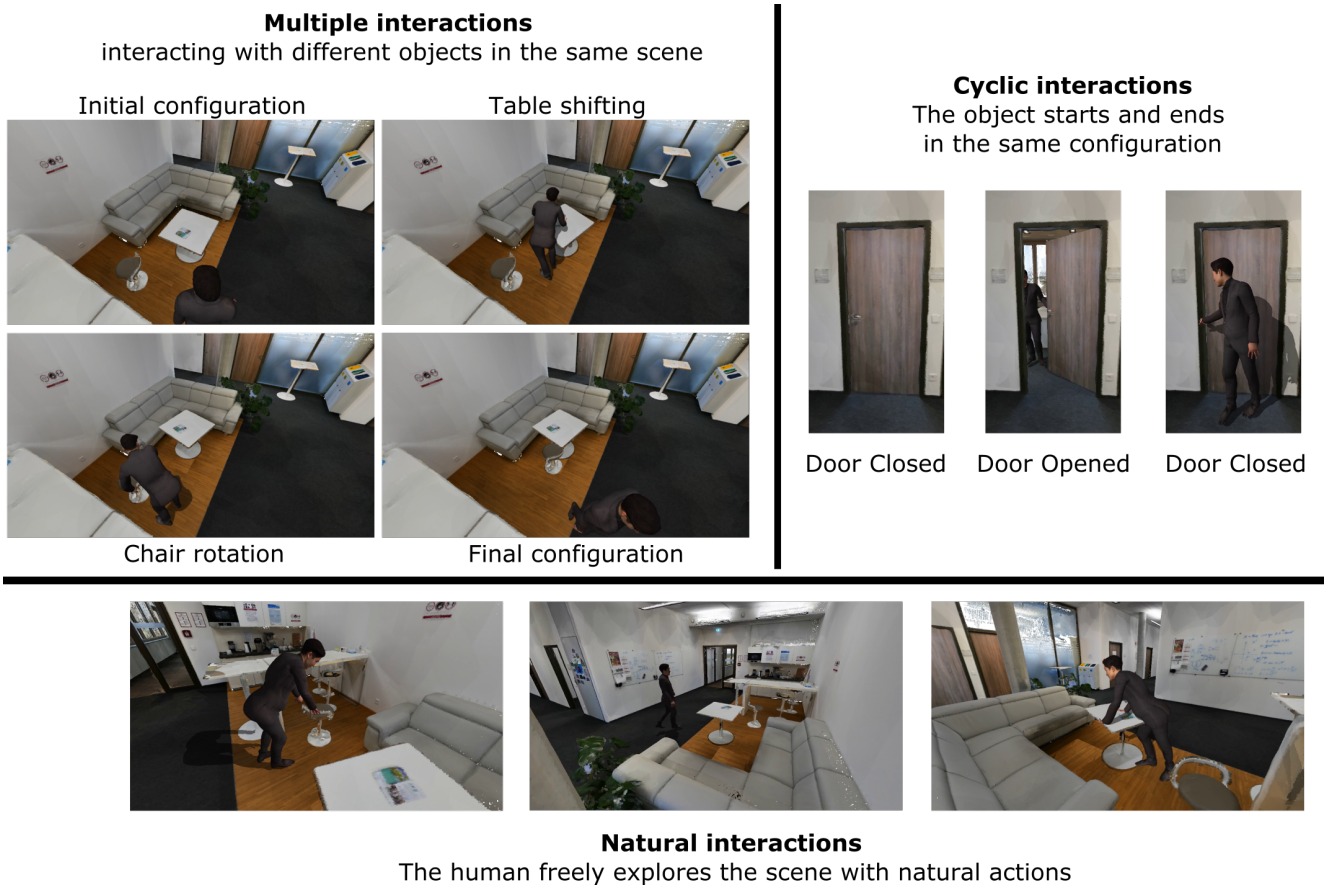

Figure 12. **Features of iReplica.** iReplica handles different kinds of challenges. It can be naturally applied to interactions involving several objects (*e.g.*, arranging a table and a chair), objects that have a cyclic behavior in the scene (*e.g.*, doors that open and close several times), and general daily interactions where the user freely moves in the space (*e.g.*, walking to the kitchen, pushing a table). Please see the supplementary video for animated results.