



HAL
open science

Towards unconstrained joint hand-object reconstruction from RGB videos

Yana Hasson, Gül Varol, Cordelia Schmid, Ivan Laptev

► **To cite this version:**

Yana Hasson, Gül Varol, Cordelia Schmid, Ivan Laptev. Towards unconstrained joint hand-object reconstruction from RGB videos. 3DV 2021 - International Conference on 3D Vision, Dec 2021, London, United Kingdom. hal-03615879

HAL Id: hal-03615879

<https://hal.science/hal-03615879v1>

Submitted on 23 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards Unconstrained Joint Hand-Object Reconstruction From RGB Videos

Yana Hasson^{1,2}

Gül Varol³

Cordelia Schmid^{1,2}

Ivan Laptev^{1,2}

¹Inria, ²Département d’informatique de l’ENS, CNRS, PSL Research University

³LIGM, École des Ponts, Univ Gustave Eiffel, CNRS, France

<https://hassony2.github.io/homan.html>

Abstract

Our work aims to obtain 3D reconstruction of hands and manipulated objects from monocular videos. Reconstructing hand-object manipulations holds a great potential for robotics and learning from human demonstrations. The supervised learning approach to this problem, however, requires 3D supervision and remains limited to constrained laboratory settings and simulators for which 3D ground truth is available. In this paper we first propose a learning-free fitting approach for hand-object reconstruction which can seamlessly handle two-hand object interactions. Our method relies on cues obtained with common methods for object detection, hand pose estimation and instance segmentation. We quantitatively evaluate our approach and show that it can be applied to datasets with varying levels of difficulty for which training data is unavailable.

1. Introduction

Joint reconstruction of hand-object interactions in the context of object manipulation has a broad range of practical applications in robotics and augmented reality. Yet, reliable and generic models for recovering 3D hand-object configurations from images do not exist to date. Learning-based approaches typically do not generalize beyond their specific training domains. Furthermore, obtaining precise 3D annotations for supervision is tedious and requires additional sensors. Existing datasets rely on depth data [10, 16], visible marker equipment [14, 52] or multiple views [6, 44, 56, 68]. Moreover, such datasets typically feature a limited number of objects [46, 57, 60, 61] while still requiring manual annotations [10] or post-processing [56, 68]. Synthetic rendering techniques can be used to create labeled datasets with a larger number of objects, however, the resulting images lack realism in terms of appearance and grasp configurations [19, 45, 67].

Developing an RGB-only method to retrieve 3D hand-object configurations for diverse objects would enable scal-

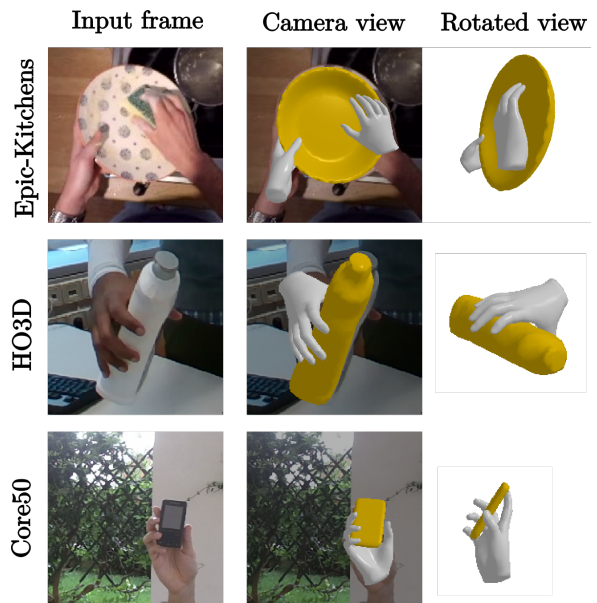


Figure 1: **Hand-object reconstructions:** We present an optimization-based method to fit 3D parametric hand model and a known object model to 2D estimates. Our method is generic and can be applied across a variety of recent video datasets presenting hand-object interactions [12, 16, 42]. Our method can also handle reconstruction of images with left and right hands.

ing up the datasets, and help the field move towards in-the-wild scenes. In this work, we argue for an optimization-based approach for its robustness across domains. Recent progress in 2D detection of objects and 3D pose estimation of isolated hands makes it possible to obtain a good initialization when fitting 3D hand-object poses to these estimates. Nevertheless, this is still very challenging due to depth ambiguities, occlusions, noisy 2D estimates and can result in physically implausible configurations. In our analysis, we show improvements over independent fits when considering a joint hand-object fitting framework with several interaction constraints. We take inspiration from

PHOSA [65], a recent work showing promising results for fitting 3D human bodies and objects, relying on object segmentation and human pose estimation models.

Several approaches propose to explore the complementarity between fitting and learning approaches for human body shape estimation. Fitting for learning has been explored for reconstructing human bodies, either semi-automatically [37] or via learning with fitting in-the-loop [30]. Recently, applying fitting on sign language videos has shown benefits for training isolated hand reconstruction models [31]. [41] shows that hand pseudo-labelling and automatic filtering using spatial and temporal consistency allows to improve the hand pose estimation branch of a model trained for joint hand-object pose estimation. Different from these works, we fit hand *and object* meshes which represent a more complex scene. [16] presents a similar approach by fitting hand-object configurations to RGB-D videos to later use in training; however, the reliance on a depth sensor significantly reduces its applicability to unconstrained images.

In this work, we propose a method which reconstructs the hand and manipulated object from short RGB video clips, assuming an approximate model for the target object is available. We investigate the strengths and limitations of our approach in challenging scenarios and present results on challenging scenes which are not currently handled by learning methods for joint hand-object pose estimation. Our code is publicly available at <https://github.com/hassony2/homan>.

Our contributions are the following: (i) We propose a fitting-based approach for hand-object reconstruction from a video clip; (ii) We present a detailed quantitative evaluation analyzing different components of our optimization method and compare to learning-based models [18] on a standard benchmark [16]; (iii) We demonstrate qualitatively the capabilities of our framework to generalize on unconstrained videos and two-hand reconstructions.

2. Related Work

We briefly review relevant literature on hand-object reconstruction, works that employ temporal constraints and hand-object datasets.

Hand-object reconstruction. 3D pose estimation for hands [22, 31, 45, 56, 67] and objects [15, 26, 36, 39, 43, 49, 63, 64] have often been tackled in isolation [38]. Joint reconstruction of hands and objects has recently received increased attention [8, 13, 16, 18, 19, 59]. Hasson et al. [19] introduces an end-to-end model to regress MANO [53] hand parameters jointly with object mesh vertices deformed from a sphere. Works of [18, 59] assume a known object model and regress a 6DOF object pose instead. Other methods focus on the grasp synthesis [11, 25, 58], or contact modeling [5, 6] given a 3D object.

Interaction constraints are imposed to avoid collisions and to encourage contact points. To achieve this, competing attraction and repulsion terms are employed in [11, 19]. Collision penalization is implemented either with approximate shape primitives [3, 34, 48] or triangle meshes [2, 16, 17, 19, 23, 33, 50, 60, 65]. Similarly, in this work we impose error terms in our joint hand-object fitting to favor physically plausible interactions.

Recent work on monocular hand-object reconstruction mostly adopts learning-based CNN models [16, 18, 19, 29, 38, 41, 59]. Such methods obtain promising results but remain limited to constrained datasets and lack generalization. Moreover, learning-based methods are typically limited to the predefined number of hands and objects. In contrast, our method can reconstruct both single and two-hand interactions detected in the image. Concurrent work of [8] extends the optimization-based body-object reconstruction method PHOSA [65] to perform hand-object fitting. While our method shares similar optimization components with [8], it differs by leveraging video data.

Temporal constraints. In case of video inputs, temporal constraints have been used for body motion estimation in the context of neural networks [20, 24], or optimization [1, 51]. For hands, [7] proposes a graph convolutional approach to learn temporal dependencies. Hampali et al. [16] make use of a temporal consistency term when fitting hand-object configurations to RGB-D data. We explore a similar term to obtain temporally smooth fits to RGB data and initialize the optimization from the previous frame’s fit. [41] propose to filter noisy hand reconstructions by detecting implausible or inconsistent hand poses and jittering predictions over consecutive frames.

3D label acquisition for hand-object datasets. Recent efforts aim to scale up data collection for 2D hand-object interactions [55]. Due to the difficulty of annotating 3D in images, several approaches for *automatic* label acquisition have been proposed. [19] introduces ObMan, a synthetic dataset obtained by rendering MANO hand model [53] automatically grasping ShapeNet objects [9]. The dataset of [14] utilizes visible magnetic sensors while [52] rely on force sensors which limit the range of possible grasps. [16] presents the HO-3D dataset where labels result from automatic fits to RGB-D data. [6] collects a dataset for 3D printed object models in a restricted capture setup. Very recently, [10] and [32] propose to collect annotations using hybrid methods to collect reliable labels. In this work, we use the HO-3D benchmark for quantitative evaluations. We also show qualitative results for videos in-the-wild where 3D ground truth is not available.

3. Automatic labelling of 3D hands and objects

We first describe the optimization-based fitting procedure, consisting of estimating 2D detections (Section 3.1),

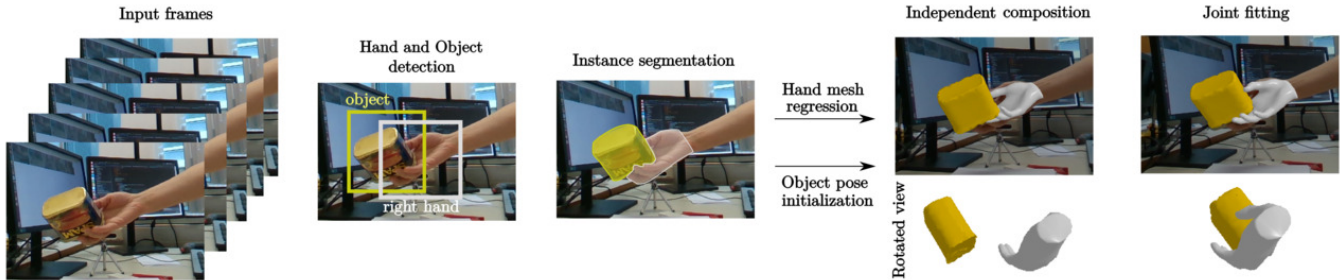


Figure 2: **Joint hand-object fitting:** We independently initialize the hand and object poses based on 2D detections and segmentations. We refine this configuration with interaction-based constraints to obtain our final joint fitting.

initializing 3D hand and object poses (Section 3.2), and joint fitting (Section 3.3). An overview can be seen in Figure 2.

Our method takes a video of hand-object manipulation as input. We assume that an exact or approximate object model representing the manipulated object is provided and use the ground-truth camera intrinsic parameters when available.

3.1. Obtaining 2D hand-object evidence

2D hand and object detection. For each video, the first step is to detect initial 2D bounding boxes. We use a recent hand and manipulated object detector [55] to extract object and hand bounding boxes in each frame. The hands are predicted with left or right side labels. When the predicted boxes do not match the known properties of the dataset in terms of object presence, hand number or hand sides, we discard the detections for the given frame, and recover detections through tracking in a subsequent step.

Tracking. We apply an off-the-shelf 2D bounding box tracker [47] which relies on Kalman filtering [35] to extract hand and object tracks from the noisy per-frame detections. This step allows to recover missed or discarded detections, and produces hand and object bounding box candidates for the full video clip.

For the Epic-Kitchens dataset, the real number of visible hands is unknown. We automatically select video clips for which at least one object and one hand track extend over more than 20 consecutive frames after tracking.

Segmentation. The key image evidence we rely on for fitting is 2D segmentation. We extract instance masks $\hat{\mathcal{M}}_{obj}$ for each tracked object detection using the instance segmentation head of the PointRend [28]. Similar to PHOSA [65], we use a model pretrained on the COCO [40] dataset. However, while [65] fits objects among the COCO categories, our target everyday objects are often not present among the COCO classes. For each object detection, we use the mask associated to the highest class activation of the PointRend instance classifier. We observe this class-agnostic approach to perform well in most cases. To account for hand occlu-

sions, we extract the COCO masks associated to the *person* class $\hat{\mathcal{M}}_{hand}$ for the tracked hand boxes, see the Section B of the supplemental material for additional details.

3.2. Independent pose initialization

Hand initialization. We employ the recent publicly available hand pose estimator FrankMocap [54] to estimate the initial hand articulated poses, as well as the hand location and scale in pixel space. We recover an estimated depth using the world scale of the hand and the exact intrinsic camera parameters when available. When the exact camera intrinsic parameters are unknown, we approximate the focal length given the specifications of the camera, and assume the central point is at the center of the pixel image.

Object initialization. We use the 2D object segmentation to initialize the object pose for the 3D model associated to the target video clip. To obtain pose candidates for the first video frame, we sample random rotations uniformly in $SO(3)$ and use the radius of the instance bounding box to estimate the 3D center of the provided mesh in the first frame. We update the object depth minimizing the difference between the *diagonals* of the projected model’s tight bounding box and the predicted detection in the least squared sense. We then compute a translation update in the camera plane, minimizing the difference between the *centers* of the projected object model and the detection given a fixed depth. We repeat these two steps until convergence to recover an estimate of the object translation. We further optimize the object pose using differentiable rendering. We optimize the object pose with a pixel squared loss on the difference between the differentially rendered object mask and the PointRend [28] object segmentation following PHOSA [65]. As for PHOSA, our object loss is hand-occlusion aware: only pixels which are not associated with the person label are taken into account. For each subsequent frame, we use the object pose from the previous frame as initialization and further refine it with differentiable rendering as described above. This process results in as many candidate motion initializations as there are candidate object

poses. In practice, the number of candidate initializations is empirically set to 50.

We select the object motion candidate for which the average *IoU* score between the rendered mask and the target occlusion mask is highest.

3.3. Joint fitting

Independent hand-object fits are often inaccurate and do not take into account interaction-based constraints. We refine the initial hand-object poses leveraging both coarse and fine-grained manipulation priors.

Optimized parameters. The goal of our fitting is to find the optimal hand and object pose parameters for a sequence of T consecutive frames. For each frame, we optimize 3D translations D_{hand}, D_{obj} , 3D global rotations R_{hand}, R_{obj} as well as θ hand pose parameters. Additionally, we optimize a shared hand scale s_{hand} .

We optimize the articulated MANO [53] model in the latent pose space θ . Given the θ pose parameters, the MANO model differentially outputs 3D hand vertex coordinates centered on the middle metacarpophalangeal joint $\mathcal{V}_{hand}^c = \text{MANO}(\theta)$. Following [58], we use a pose latent subspace of size 16.

We optimize the hands and object rotation R_{hand}, R_{obj} using the 6D continuous rotation representation [66] and optimize the 3D translation D_{hand}, D_{obj} in metric camera space. When we use approximate object meshes, we additionally optimize a scalar scaling parameter s_{obj} which allows the object’s size to vary. We also allow hand vertices to scale by a factor s_{hand} which is shared across the T frames. The hand vertices in camera coordinates \mathcal{V}_{hand}^{3d} are estimated as following:

$$\mathcal{V}_{hand}^{3d} = s_{hand}(R_{hand}\mathcal{V}_{hand}^c) + D_{hand}. \quad (1)$$

The object vertices \mathcal{V}_{obj}^{3d} are estimated as a rigid transformation of canonically oriented model vertices \mathcal{V}_{obj}^c :

$$\mathcal{V}_{obj}^{3d} = s_{obj}(R_{obj}\mathcal{V}_{obj}^c) + D_{obj}. \quad (2)$$

Next, we describe the individual error terms that we minimize during fitting.

Object silhouette matching (\mathcal{L}_{obj}). We use a differentiable renderer [26] to render the object mask \mathcal{M}_{obj} and compare it to the reference segmentation mask $\hat{\mathcal{M}}_{obj}$. This error term is occlusion-aware as in [65]. No penalization occurs for the object silhouette being rendered in pixel regions $\hat{\mathcal{M}}_{hand}$ where hand occlusions occur. We write this error as:

$$\mathcal{L}_{obj} = \|(1 - \hat{\mathcal{M}}_{hand}) \circ (\mathcal{M}_{obj} - \hat{\mathcal{M}}_{obj})\|_2^2 \quad (3)$$

Projected hand vertices (\mathcal{L}^{v2d}). We constrain the hand position by penalizing projected vertex offsets from the initial vertex pixel locations \mathcal{V}_{2d}^{hand} predicted by FrankMocap [54].

To compute the current 2D vertex locations, we project the MANO [53] vertices \mathcal{V}_{hand}^{3D} to the pixel plane using the camera projection operation Π . This error is written as:

$$\mathcal{L}^{v2d} = \|\Pi(\mathcal{V}_{hand}^{3D}) - \hat{\mathcal{V}}_{hand}^{2d}\|_2^2 \quad (4)$$

Hand regularization (\mathcal{L}_{pca}). Given that we optimize the articulated hand pose, we regularize the optimized hand pose. As in [4, 19, 54], we apply a mean square error term to the PCA hand components $\mathcal{L}_{pca} = \|\theta\|_2^2$ to bias the estimated hand poses towards statistically plausible configurations.

Scale (\mathcal{L}_{scale}). Similarly to PHOSA[65], when we allow the elements in the scene to scale, we penalize deviations from category-level average dimensions.

Smoothness (\mathcal{L}_{smooth}). We further leverage a simple smoothness prior over the T sampled frames $\mathcal{L}_{smooth} = \sum_{t=1}^{T-1} \|\mathcal{V}_{t+1}^{3D} - \mathcal{V}_t^{3D}\|_2^2$ which encourages minimal 3D vertex variances across neighboring frames for both hands and objects as in [16].

Coarse interaction ($\mathcal{L}_{centroid}$). Following [65], we penalize the squared distance between hand and object centroids when the predicted hand and object boxes overlap to encode a coarse interaction prior. As we assume the object scale to be provided, this error only impacts the rigid hand pose, effectively attracting the hand towards the interacted object. In case of multiple hands, all overlapping hand-object pairs of meshes are considered.

Collision (\mathcal{L}_{col}). We rely on a recent collision penalization term introduced to enforce non-interpenetration constraints between multiple persons in the context of body mesh estimation [23]. The collision error \mathcal{L}_{col} is computed for each pair k, l of estimated meshes. We compute $L_{col}^{k,l} = \sum_i \Phi_k(\mathcal{V}_i^l)$. Where Φ_k is the negative truncated signed distance function (SDF) associated to the mesh k , $\Phi_k(\mathcal{V}) = \max(0, -SDF(\mathcal{V}))$.

$$\mathcal{L}_{col} = \sum_{k,l} \mathcal{L}_{col}^{k,l} \quad (5)$$

This formulation allows to handle any number of visible hands and objects in the scene.

Local contacts. Hands interact with objects by establishing surface contacts without interpenetration. We experiment with the hand-object heuristic introduced by [19]. We repurpose this loss which has been introduced in a learning framework to our optimization setup. This additional term encourages the contacts to occur at the surface of the object by penalizing the distance between hand vertices the closest object vertex for hand vertices in the object’s vicinity. We refer to the supplemental material Section C for additional details.

The final objective \mathcal{L} is composed of a weighted sum of the previously described terms, where the weights are

empirically set to balance the contributions of each error term.

$$\begin{aligned} \mathcal{L} = & \lambda_{obj} \mathcal{L}_{obj} + \lambda_{v2d} \mathcal{L}_{v2d} \\ & + \lambda_{pca} \mathcal{L}_{pca} + \lambda_{scale} \mathcal{L}_{scale} + \lambda_{smooth} \mathcal{L}_{smooth} \\ & + \lambda_{centroid} \mathcal{L}_{centroid} + \lambda_{local} \mathcal{L}_{local} + \lambda_{col} \mathcal{L}_{col} \end{aligned} \quad (6)$$

While [65] adapt the weights for their optimization for each object categories, we fix the weight parameters empirically and keep them constant across all experiments. We refer to Section C of the supplemental material for exact weight values and additional implementation details.

4. Experiments

We first define the evaluation metrics (Section 4.1) and the datasets (Section 4.2) used in our experiments. Then, we provide an ablation to measure the contribution of each of our optimization objective terms (Section 4.3). We investigate the sensitivity of our approach to the quality of the 2D estimates (Section 4.4). Next, we compare our approach to the state of the art (Section 4.5). Finally, we provide qualitative results for in-the-wild examples (Section 4.6).

4.1. Metrics

The structured output for hand-object reconstruction is difficult to evaluate with a single metric. We therefore rely on multiple evaluation measures.

Object metrics. We evaluate our object pose estimates by computing the average vertex distance. Common objects such as bottles and plates often present plane and revolution symmetries. To account for point matching ambiguities, we further report the standard pose estimation average closest point distance (add-s) [64].

Hand metrics. We follow the standard hand pose estimation protocols [16, 68] and report the procrustes-aligned hand vertex error and F-scores. We compare the hand joint predictions using average distances after scale and translation alignment. When investigating the results of the joint fitting, we additionally report the average hand vertex distances without alignment.

Interaction metrics. *Penetration depth (mm):* We report the maximum penetration depth between the hand and the object following previous work on hand-object interactions [6, 18, 19, 25]. *Contact (%):* We also report the contact percentage following [25]. When ground truth contact binary labels are available, we report contact accuracy, as we further detail in the supplemental material Section C.

4.2. Datasets

HO-3D [16] is the reference dataset which provides accurate hand-object annotations during interaction for markerless RGB images. The users manipulate 10 objects from

the YCB [69] dataset, for which the CAD models are provided. The ground-truth annotations are obtained by fitting the hand and object models to RGB-D evidence which assumes limited hand motion. We present results on the standard test set composed of 13 videos with a total of 11525 frames depicting single-hand object manipulations.

Core50 [42] contains short sequences of unannotated images of hands manipulating 50 object instances from 10 everyday object categories such as cups, light bulbs and phones. We manually associate 26 objects with approximately matching 3D object models from the ShapeNet dataset [9]. We further annotate hands being left or right for each of the 11 video sequences available for each object, resulting in 286 video clips and 86k frames.

Epic Kitchens [12] is an unscripted dataset which has been collected without imposing constraints or equipment beyond a head-mounted camera. In contrast to existing datasets for which 3D information is available, it therefore presents natural hand-object interactions. This dataset is however densely annotated with action labels which include the category of the object of interest. We focus on a subset of common object categories: cups, plates, cans, phones and bottles which are involved in a total of 3456 action video clips. For each object category, we associate an object model from the CAD ShapeNet [9] database. We assume that the target manipulated object is the one with the longest track in the associated action clip.

4.3. Contribution of error terms in fitting

As explained in Section 3, our method introduces several error terms which determine the final reconstruction quality. We evaluate the contribution of the main error terms on the HO-3D benchmark [16] in Table 1. We validate that our joint reconstruction outperforms the naive composition baseline, which is obtained by separately fitting the object to the occlusion-aware object mask and the hand using the hand-specific terms \mathcal{L}_{v2d} and \mathcal{L}_{pca} . When fitted independently, the scale-depth ambiguity prevents an accurate estimate of the hand distance. As we use the ground-truth object model, the 3D object pose can be estimated without ambiguity using the camera intrinsic parameters. Joint fitting improves the 3D pose estimates using both smoothness and interaction priors. We observe that the coarse interaction prior is critical towards improving the absolute hand pose. When removing this error term, the hand pose error increases two-fold from 8.9 to 17.1cm. We observe that the temporal smoothness term, while simple, provides a strong improvement to both the hand and object pose estimates. Leveraging information across neighboring video frames reduces the errors by 25% and 38% for the hand and object, respectively. While the local interaction and collision penalization terms only marginally change the hand and object reconstruction scores, their impact can be quan-

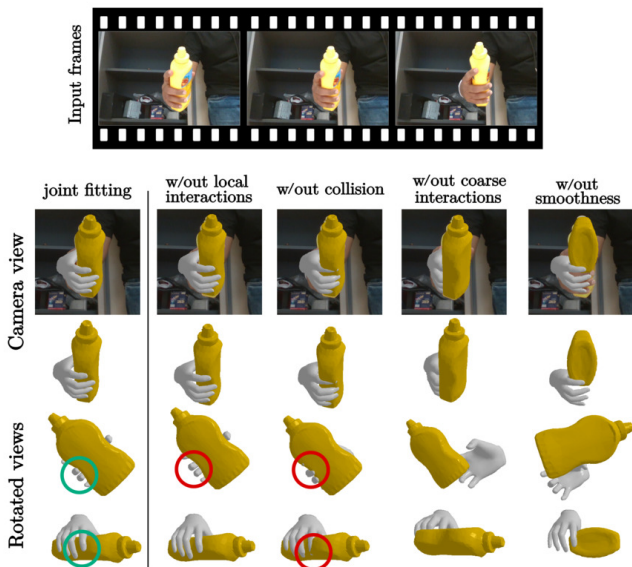


Figure 3: **Effect of error terms:** Qualitative analysis showing the effects of the various error terms for the hand-object reconstruction accuracy on the HO-3D dataset. We highlight visual evidence of local corrections attributed to the local interaction [19] and collision [23] terms.

tatively observed in the interaction metrics. The collision penalization terms reduce the average penetration depth by a large factor (10.2mm vs 2.4mm). The local interaction term [19] reduces both the interpenetration depth and the contact percentage, which is defined as either exact surface contact or interpenetration between the hand and the object. Qualitatively, we observe that this term produces local corrections in the vicinity of estimated contact points. Including all error terms results in more plausible grasps, which we illustrate in Figure 3 qualitatively.

While ground truth 3D poses are hard to annotate for generic videos with hand-object manipulations, interaction metrics such as penetration depth can be directly computed from the predicted reconstructions. As the Core50 [42] dataset presents only videos in which the object is actively manipulated by the hand, we can additionally report contact accuracy as proxies to evaluate the quality of the reconstructed grasps. We report these two metrics on the Core50 dataset in Table 2 and confirm the benefit from our joint fitting approach. Using joint fitting significantly increases the contact accuracy from 7.3% to 89.5%, while only increasing the average penetration depth by 0.6mm.

4.4. Sensitivity to estimated 2D evidence

Our method makes use of generic models for detection and mask estimation and would directly benefit from more



Figure 4: **Sensitivity to 2D detections:** Dependence of our 3D reconstruction on the accuracy of the 2D evidence by running our method with ground truth (GT) hand and object detections and ground truth object masks for the HO-3D dataset [16].

accurate detection, 2D hand pose estimation and instance segmentation models. The reliance of our method on 2D cues therefore allows it to benefit from additional efforts in 2D image annotation which is simpler compared to 3D annotation in practice. We investigate the dependence of our method on the quality of the available 2D evidence. To investigate the expected improvements our method could gain from stronger object detections, instead of using noisy detections, we use the hand and object ground truth bounding boxes provided for the HO-3D test set. We observe in Figure 4 the improvements we obtain from using the ground-truth detections. Both hands and objects benefit from more accurate detections, improving by 2cm when compared to the tracking-by-detection estimates. To investigate the errors which come from using noisy approximate instance masks, we render the object and hand ground truth masks and use them to guide our optimization. By relying on 2D information, our approach suffers from limitations such as depth ambiguities which can result from fitting to image segmentation masks. Object asymmetries which rely on color information can also be hard to resolved during fitting. We observe that using ground truth hand and object masks allows to further decrease the 3D pose errors. We note that the object error decreases to 2cm and below 1cm when comparing distances to closest points. When the object model is available, our joint fitting method produces highly accurate object poses in the presence of accurate 2D evidence.

4.5. State-of-the-art comparison

Recent efforts for joint hand-object pose estimation in camera space [18, 19, 59] have focused on direct bottom-up regression of 3D poses. We compare the performance of our fitting approach to recent learning-based methods for joint hand-object reconstruction [18, 41] in Table 3.

While these learnt methods produce more accurate hand predictions, their object predictions are *instance specific*. As a direct consequence, the methods do not generalize to

¹<https://competitions.codalab.org/competitions/22485>

	Hand		Object		Interaction	
	vertex mean distance (cm)↓	mepe aligned (cm) ↓	vertex mean distance (cm) ↓	add-s (cm) ↓	pen. depth (mm) ↓	contact %
indep. composition	26.2	5.2	12.1	7.7	3.2	25.8
joint fitting	8.6	5.4	8.0	3.8	2.8	77.5
w/out local interactions	8.5	5.4	8.0	3.8	2.4	72.3
w/out collision	8.9	5.4	8.0	3.8	10.2	80.5
w/out coarse interaction	17.1	5.3	8.1	3.8	1.9	59.4
w/out smoothness	11.4	5.6	12.8	8.3	3.0	79.1

Table 1: **Contribution of error terms:** We show benefits of the *joint* modeling for hand-object interactions by the increased reconstruction accuracy when compared to independent hand and object composition on the HO-3D [16] dataset. Our smoothness and interaction terms impose additional constraints which improve the final hand-object pose reconstructions.

Dataset	Contact Accuracy (%) ↑		Pen. Depth (mm) ↓	
	independent	joint	independent	joint
Core50	7.3	89.5	0.6	1.2

Table 2: **Results on Core50:** Interaction errors for hand-object fits obtained on the Core50 dataset. We observe significantly improved contact accuracy with joint fitting over independent fits at the expense of a minor cost of a 0.6mm increase in penetration.

Method	mesh error ↓	F-score @5mm ↑	F-score F@15mm ↑	generalizes to unseen objects
Liu 2021 [41]	0.95	0.96	0.53	✗
Hampali 2020 [16]	1.06	0.51	0.94	✗
Hasson 2020 [18]	1.14	0.42	0.93	✗
Joint fitting	1.47	0.39	0.88	✓

Table 3: **State-of-the-art-comparison:** We compare the hand performance of the single-view pose estimation baselines. Note that the reported results for [16] output hand meshes only. Hasson 2020 [18], Liu 2021 [41] and our method predict the hand-object meshes jointly. All methods are trained only on the real images from the HO-3D training split and evaluated on the official test split through an online submission¹. The hand mesh error is reported after procrustes alignment.

new objects at test time. In contrast, our generic fitting method performs equally well across the seen and unseen objects of the HO-3D test split, see Table 4.

4.6. In-the-wild 3D hand-object pose estimation

We test the limits of our approach and showcase the strength of our method by comparing qualitatively to a model trained for joint hand-object reconstruction [19]. Their model estimates the shape of the object by deforming a sphere and therefore does not depend on known ob-

	Object				Hand	
	vertex dist (cm) ↓		add-s (cm) ↓		mepe ↓	aligned mepe ↓
	Seen	Unseen	Seen	Unseen		
Ours [18]	8.0	8.1	4.0	3.3	8.6	5.4
	6.7	10.7	2.2	3.6	5.5	3.7

Table 4: **Unseen objects:** Vertex errors (cm) for estimated hand and object meshes. Compared to [18], our method performs similarly across seen and *unseen* objects.

ject models. However, given this object topology restriction, this method has limited expressivity. It can only capture a subset of all object shapes which excludes everyday objects such as mugs or cups, see Fig. 6. In comparison, while our method makes stronger assumptions by relying on an approximate object model, it is applicable to any everyday objects for which an approximate mesh can be retrieved without further limitations. Additionally, while the manipulation reconstruction from [19] estimates the grasp relative to the root joint of the hand, our method outputs image-aligned predictions.

We further show that our method can be applied to the challenging Epic-Kitchens dataset [12] which presents natural manipulation of common objects, see Fig. 5. Note that our objective (6) is not restricted to a single hand-object pair and naturally generalizes to multiple hands and objects. To handle scenes with two hands in [12] we optimize (6) with pairwise losses defined for both detected hands and the detected object. We show results of two-hand manipulations which represent the majority of examples in the target dataset. While we observe cases of depth ambiguity, especially with almost planar objects such as plates, we show that our method can recover plausible reconstructions for several object categories across a variety of hand poses.

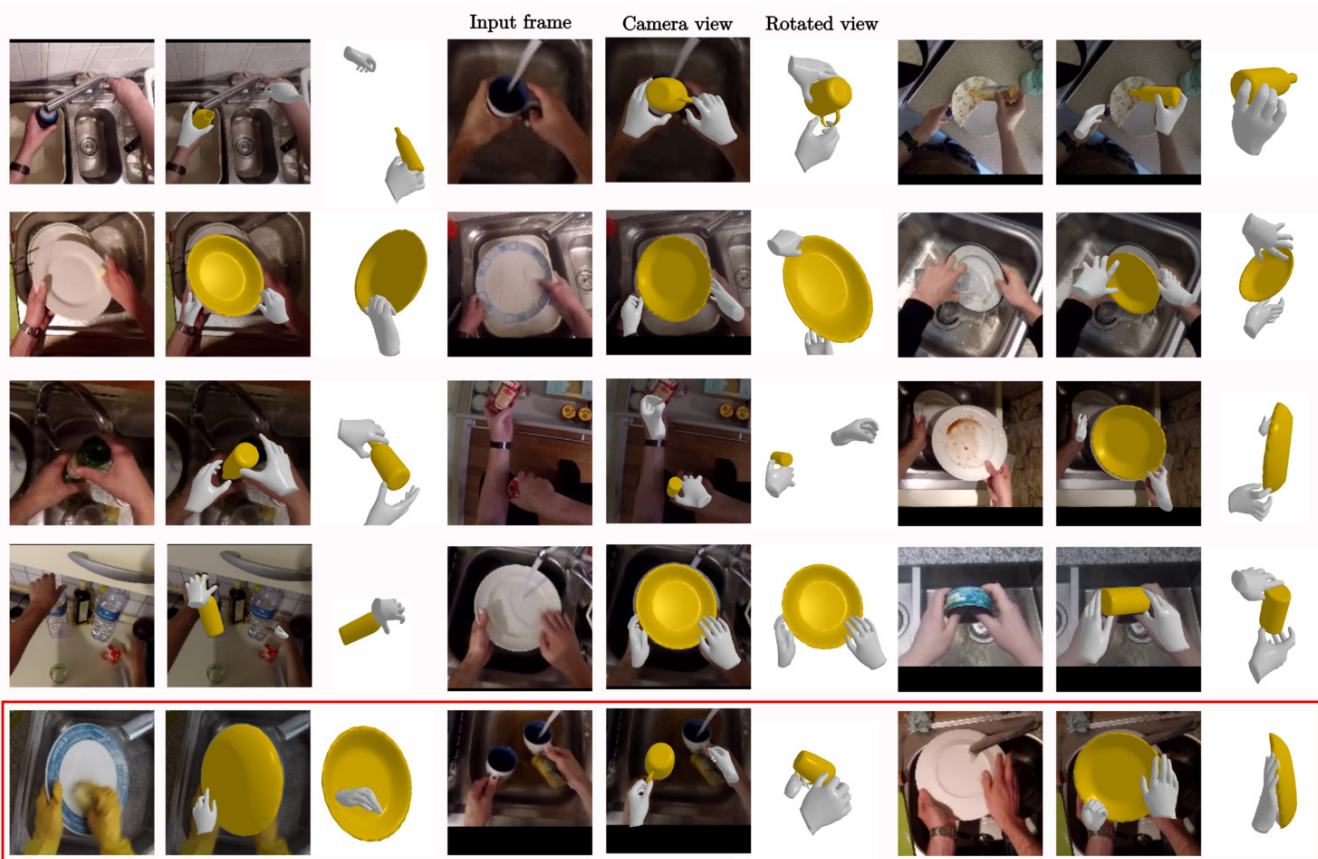


Figure 5: **In-the-wild reconstructions:** Our results on natural hand-object manipulations of the Epic-Kitchens dataset [12]. We present several success and failures of our method on the challenging Epic-Kitchens dataset. We highlight typical failure modes for our method, in particular, object orientation errors resulting from depth ambiguity. We observe that our fitting method recovers plausible interactions across different object categories and hand-object configurations.



Figure 6: **Comparison with [19]:** Qualitative comparison of our fits to the ObMan-trained model [19] estimations on the Core50 dataset. While our model requires an approximate mesh to be provided, it generalizes to objects of arbitrary topology.

5. Conclusions

We present a robust approach for fitting 3D hand-object configurations to monocular RGB videos. Our method

builds on estimates from neural network models for detection, object segmentation and 3D hand pose estimation trained with full supervision. Due to the lack of supervision at similar scale for 3D hand-object interactions, we opt for a fitting-based approach and demonstrate advantages on several datasets. A key limitation of current methods that estimate 6DOF object pose is the reliance on known object models. Future work will consider automatic object recovery to fully automate the hand-object reconstruction process.

Acknowledgments. This work was funded in part by the MSR-Inria joint lab, the French government under management of Agence Nationale de la Recherche as part of the “Investissements d’avenir” program, reference ANR19-P3IA-0001 (PRAIRIE 3IA Institute), by Louis Vuitton ENS Chair on Artificial Intelligence, as well as gifts from Google and Adobe. This work was granted access to the HPC resources of IDRIS under the allocation 2020-AD011012004R1 made by GENCI.

References

- [1] Anurag* Arnab, Carl* Doersch, and Andrew Zisserman. Exploiting temporal context for 3D human pose estimation in the wild. In *CVPR*, 2019.
- [2] Luca Ballan, Aparna Taneja, Juergen Gall, Luc Van Gool, and Marc Pollefeys. Motion capture of hands in action using discriminative salient points. In *ECCV*, 2012.
- [3] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, V. Peter Gehler, Romero Javier 0002, and J. Michael Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *ECCV*, 2016.
- [4] Adnane Boukhayma, Rodrigo de Bem, and Philip H.S. Torr. 3D hand shape and pose from images in the wild. In *CVPR*, 2019.
- [5] Samarth Brahmabhatt, Cusuh Ham, Charles C. Kemp, and James Hays. ContactDB: Analyzing and predicting grasp contact via thermal imaging. In *CVPR*, 2019.
- [6] Samarth Brahmabhatt, 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.
- [7] Yujun Cai, Liuhao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. Exploiting spatial-temporal relationships for 3D pose estimation via graph convolutional networks. In *ICCV*, 2019.
- [8] Zhe Cao, Ilija Radosavovic, Angjoo Kanazawa, and Jitendra Malik. Reconstructing hand-object interactions in the wild. *ICCV*, 2021.
- [9] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An information-rich 3D model repository. *arXiv:1512.03012*, 2015.
- [10] Yu-Wei Chao, Wei Yang, Yu Xiang, Pavlo Molchanov, Ankur Handa, Jonathan Tremblay, Yashraj S. Narang, Karl Van Wyk, Umar Iqbal, Stan Birchfield, Jan Kautz, and Dieter Fox. DexYCB: A benchmark for capturing hand grasping of objects. In *CVPR*, 2021.
- [11] Enric Corona, Albert Pumarola, Guillem Alenya, Francesc Moreno-Noguer, and Gregory Rogez. GanHand: Predicting human grasp affordances in multi-object scenes. In *CVPR*, 2020.
- [12] 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.
- [13] Bardia Doosti, Shujon Naha, Majid Mirbagheri, and David Crandall. Hope-net: A graph-based model for hand-object pose estimation. In *CVPR*, June 2020.
- [14] 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.
- [15] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry. AtlasNet: A papier-mâché approach to learning 3D surface generation. In *CVPR*, 2018.
- [16] Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. HONotate: A method for 3D annotation of hand and object poses. In *CVPR*, 2020.
- [17] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J. Black. Resolving 3D human pose ambiguities with 3D scene constraints. In *ICCV*, 2019.
- [18] 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*, 2020.
- [19] Yana Hasson, Gül Varol, Dimitris Tzionas, Igor Kalevatykh, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *CVPR*, 2019.
- [20] Mir Rayat Imtiaz Hossain and James J. Little. Exploiting temporal information for 3D human pose estimation. In *ECCV*, 2018.
- [21] Jingwei Huang, Yichao Zhou, and Leonidas Guibas. Manifoldplus: A robust and scalable watertight manifold surface generation method for triangle soups. *arXiv:2005.11621*, 2020.
- [22] Umar Iqbal, Pavlo Molchanov, Thomas Breuel, Juergen Gall, and Jan Kautz. Hand pose estimation via latent 2.5d heatmap regression. In *ECCV*, 2018.
- [23] Wen Jiang, Nikos Kolotouros, Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Coherent reconstruction of multiple humans from a single image. In *CVPR*, 2020.
- [24] Angjoo Kanazawa, Jason Y. Zhang, Panna Felsen, and Jitendra Malik. Learning 3D human dynamics from video. In *CVPR*, 2019.
- [25] Korrawe Karunratanakul, Jinlong Yang, Yan Zhang, Michael Black, Krikamol Muandet, and Siyu Tang. Grasping field: Learning implicit representations for human grasps. In *3DV*, 2020.
- [26] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3D mesh renderer. In *CVPR*, 2018.
- [27] Diederick P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.
- [28] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. PointRend: Image segmentation as rendering. *arXiv:1912.08193*, 2019.
- [29] Mia Kokic, Danica Kragic, and Jeannette Bohg. Learning task-oriented grasping from human activity datasets. *IROS*, 2020.
- [30] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In *ICCV*, 2019.
- [31] Dominik Kulon, Riza Alp Güler, I. Kokkinos, M. Bronstein, and S. Zafeiriou. Weakly-supervised mesh-convolutional hand reconstruction in the wild. In *CVPR*, 2020.
- [32] Taein Kwon, Bugra Tekin, Jan Stuhmer, Federica Bogo, and Marc Pollefeys. H2o: Two hands manipulating objects for first person interaction recognition. *arXiv:2104.11181*, 2021.
- [33] Nikolaos Kyriazis and Antonis A. Argyros. Physically plausible 3D scene tracking: the single actor hypothesis. In *CVPR*, 2013.
- [34] Nikolaos Kyriazis and Antonis A. Argyros. Scalable 3D tracking of multiple interacting objects. In *CVPR*, 2014.
- [35] Roger Labbe. Kalman and bayesian filters in python. <https://github.com/rllabbe/Kalman-and-Bayesian-Filters-in-Python>, 2014.

- [36] Y. Labbe, J. Carpentier, M. Aubry, and J. Sivic. CosyPose: Consistent multi-view multi-object 6D pose estimation. In *ECCV*, 2020.
- [37] Christoph Lassner, Javier Romero, Martin Kiefel, Federica Bogo, Michael J. Black, and Peter V. Gehler. Unite the people: Closing the loop between 3d and 2d human representations. In *CVPR*, 2017.
- [38] Vincent Lepetit. Recent advances in 3d object and hand pose estimation. *arXiv:2006.05927*, 2020.
- [39] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. Deepim: Deep iterative matching for 6d pose estimation. In *ECCV*, 2018.
- [40] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and Larry Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.
- [41] Shaowei Liu, Hanwen Jiang, Jiarui Xu, Sifei Liu, and Xiaolong Wang. Semi-supervised 3d hand-object poses estimation with interactions in time. In *CVPR*, 2021.
- [42] Vincenzo Lomonaco and Davide Maltoni. Core50: a new dataset and benchmark for continuous object recognition. In *Proceedings of the 1st Annual Conference on Robot Learning*. PMLR, 2017.
- [43] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3D reconstruction in function space. In *CVPR*, 2019.
- [44] Gyeongsik Moon, Shou-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. In *ECCV*, 2020.
- [45] Franziska Mueller, Florian Bernard, Oleksandr Sotnychenko, Dushyant Mehta, Srinath Sridhar, Dan Casas, and Christian Theobalt. GANerated hands for real-time 3D hand tracking from monocular RGB. In *CVPR*, 2018.
- [46] 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 *ICCV*, 2017.
- [47] Wiktor Muron. motpy. <https://github.com/wmuron/motpy>.
- [48] Iasonas Oikonomidis, Nikolaos Kyriazis, and Antonis A. Argyros. Full DOF tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *ICCV*, 2011.
- [49] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *CVPR*, 2019.
- [50] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *CVPR*, 2019.
- [51] Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. Sfv: Reinforcement learning of physical skills from videos. *ACM Trans. Graph.*, 2018.
- [52] Tu-Hoa Pham, Nikolaos Kyriazis, Antonis A. Argyros, and Abderrahmane Kheddar. Hand-object contact force estimation from markerless visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.
- [53] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), 2017.
- [54] Yu Rong, Takaaki Shiratori, and Hanbyul Joo. FrankMocap: Fast monocular 3D hand and body motion capture by regression and integration. *arXiv:2008.08324*, 2020.
- [55] Dandan Shan, Jiaqi Geng, Michelle Shu, and David Fouhey. Understanding human hands in contact at internet scale. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.
- [56] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *CVPR*, 2017.
- [57] 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.
- [58] Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas. GRAB: A dataset of whole-body human grasping of objects. In *ECCV*, 2020.
- [59] Bugra Tekin, Federica Bogo, and Marc Pollefeys. H+O: Unified egocentric recognition of 3D hand-object poses and interactions. In *CVPR*, 2019.
- [60] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. Capturing hands in action using discriminative salient points and physics simulation. *IJCV*, 2016.
- [61] Dimitrios Tzionas and Juergen Gall. 3D object reconstruction from hand-object interactions. In *ICCV*, 2015.
- [62] Sébastien Valette, Jean-Marc Chassery, and Rémy Prost. Generic remeshing of 3d triangular meshes with metric-dependent discrete voronoi diagrams. *Transactions on Visualization and Computer Graphics*, 2008.
- [63] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2Mesh: Generating 3D mesh models from single RGB images. In *ECCV*, 2018.
- [64] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. PoseCNN: A convolutional neural network for 6d object pose estimation in cluttered scenes. In *Robotics: Science and Systems (RSS)*, 2018.
- [65] 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 *ECCV*, 2020.
- [66] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *CVPR*, 2019.
- [67] Christian Zimmermann and Thomas Brox. Learning to estimate 3D hand pose from single RGB images. In *ICCV*, 2017.
- [68] 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.
- [69] Berk Çalli, Arjun Singh, and Aaron Walsman. The ycb object and model set: Towards common benchmarks for manipulation research. In *ICAR*, 2015.

APPENDIX

We first present additional qualitative results for the HO-3D [16] dataset (Section A). Next, we analyze the 2D detection and segmentation performance of the general-purpose estimators on the HO-3D [16] dataset (Section B). Finally, we provide additional implementation details for our joint hand-object optimization (Section C).

A. Qualitative results for HO-3D dataset

In Figure A.2, we compare qualitatively on the HO-3D test set our joint fitting method with independent composition, which results from fitting the object and the hand without interaction constraints independently on each frame. We observe that the joint fitting allows to recover improved configurations across various test objects and consistently improves upon independent composition.

B. 2D evidence accuracy.

B.1. Evaluation of hand and object detection

We assess the performance of our procedure to recover the detections and segmentation masks for the hands and manipulated objects.

We use the 100 Days of Hands (100DOH) hand-object detector [55] to predict hand and object bounding boxes. Following [55], we evaluate hand detection accuracy by reporting average precision (AP) for predicted hand and object boxes and set the intersection over union (IoU) threshold to 0.5 on the HO-3D [16] train split. To compute ground truth detections, we compute the tight bounding box around pixel coordinates of the projected vertices for the objects and the hand. For the hands, we post-process the detections to recover square boxes. In Table. A.1 we report the AP for hands and objects before and after tracking. We observe that tracking [47] effectively allows us to recover valid detections. The use of tracking improves AP by 24% and 12% for hands and objects respectively.

B.2. Evaluation of hand and object segmentation

We evaluate the accuracy of our instance mask estimation procedure. We report the IoU between the

	Hand AP	Object AP
Without tracking	0.61	0.59
With tracking	0.85	0.71

Table A.1: **Detection accuracy:** Hand and object detection precision using the 100DOH detector on the HO-3D [16] train split. Tracking allows to retrieve reliable detections both for hands and objects.

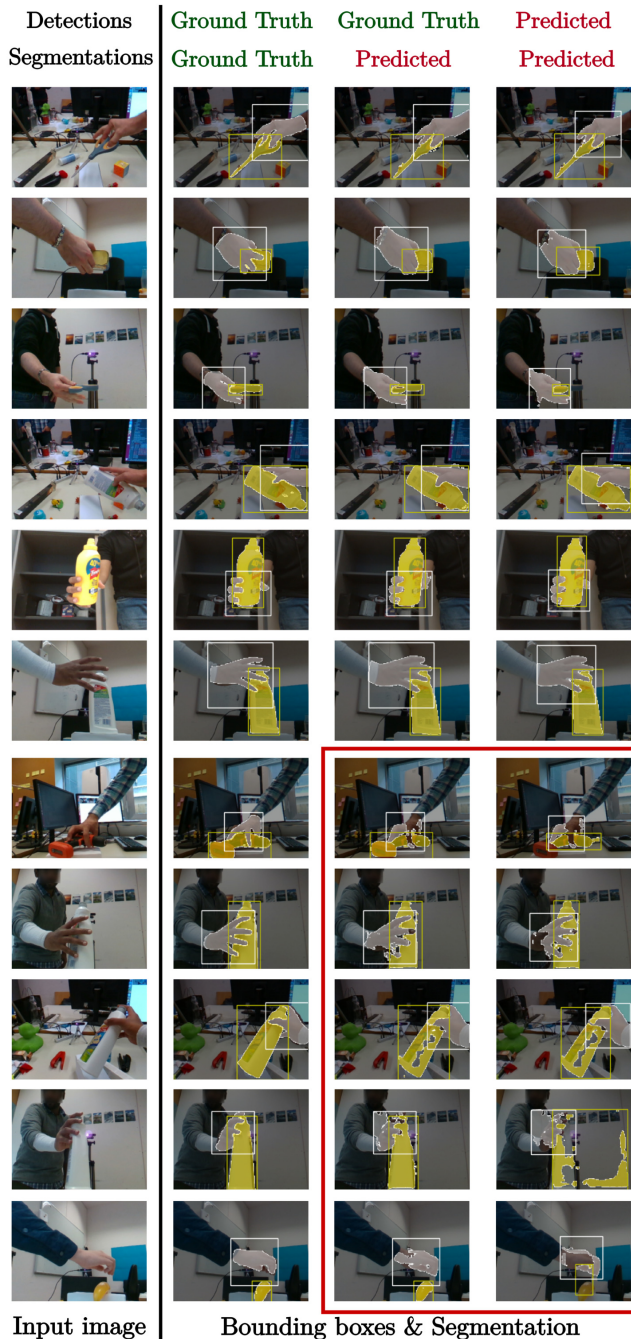


Figure A.1: We qualitatively investigate the performance of the 2D detection and segmentation models on the HO-3D [16] train split. Outlined in red, we display segmentation failures for hands or objects which are either due to imprecise detections or occur despite good detections.

PointRend [28] outputs and the ground truth segmentation masks, which we obtain by rendering the ground truth

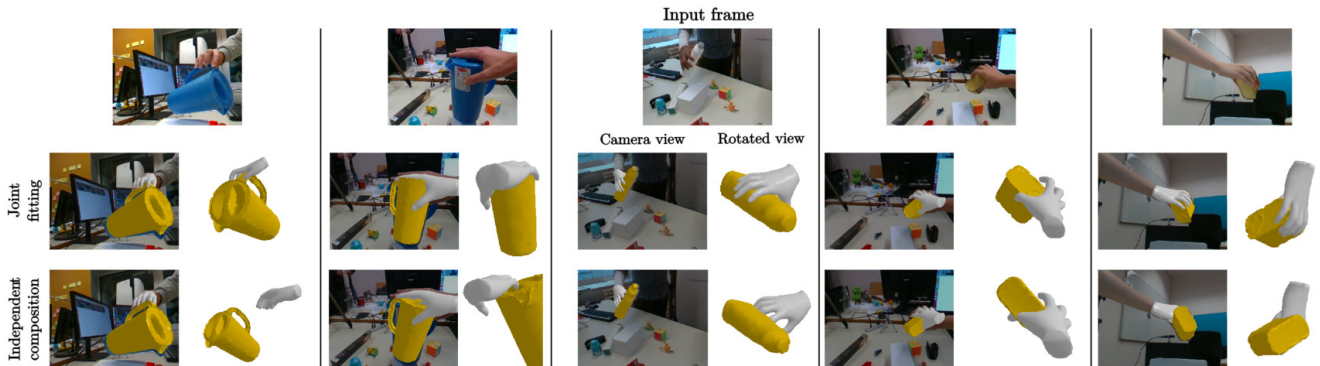


Figure A.2: Reconstructions obtained from independent composition and joint fitting on HO-3D [16] dataset. We observe that enforcing hand-object interaction constraints allows our method to recover plausible grasps and significantly improves upon the naive independent composition baseline.

Detections	Hand IoU	Object IoU
Ground Truth boxes	0.63	0.74
Predicted boxes [55]	0.61	0.66

Table A.2: **Segmentation accuracy:** Hand and object segmentation IoU for the PointRend mask estimates on the HO-3D dataset [16].

meshes. As the hand ground truth meshes are unavailable for the test split, we report results on the HO-3D train split. As described in the method section of the main paper, we select the mask produced for the most confident class prediction for the object and the Person class for the hand. To investigate the segmentation error which is due to bounding box errors, we report IoU scores for the PointRend segmentation head both using the tracked and the ground truth detections in Table A.2. As expected, segmentation masks are more precise when using ground truth object boxes, especially for objects. Hence, our method would benefit from more robust and accurate detections. We illustrate success and failures for 2D mask extraction in Figure A.1.

C. Implementation details.

We describe details of our fitting implementation. Our code and manually collected annotations will be made available upon publication.

Optimization procedure. Two of our error terms, the collision and fine-grained interactions act locally and require a reasonable initial configuration to provide meaningful gradients. We therefore obtain the final hand-object reconstructions by performing two consecutive steps. We first recover a coarse pose for the hand and the object using the data and coarse interaction and regularization terms (\mathcal{L}_{obj} , \mathcal{L}_{v2d} , \mathcal{L}_{pca} , \mathcal{L}_{scale} , \mathcal{L}_{smooth} , $\mathcal{L}_{centroid}$). We then refine the optimized parameters using the full set of error

terms. Both steps are run for 200 optimization steps using the Adam [27] optimizer. We use different learning rates for various optimized parameters in order to distribute the updates between articulated and rigid pose updates. We use a fixed learning rate of 0.1 for the PCA components of the MANO [53] articulated hand model and the continuous rotation parameters [66]. For the translation and scale updates we use a learning rate of 0.01.

Error terms weighting. We use the same weights to balance the various error terms across all reconstructed datasets. We empirically fix the weights to the following values.

Data weights: $\lambda_{obj} : 1$, $\lambda_{v2d} : 50$

Regularization weights: $\lambda_{pca} : 0.004$, $\lambda_{scale} : 0.001$, $\lambda_{smooth} : 2000$

Interaction weights: $\lambda_{centroid} : 1$, $\lambda_{local} : 1$, $\lambda_{col} : 0.001$

Mesh preprocessing. In order to reason about collisions and compute penetration depths, we preprocess the meshes to make them watertight. We further uniformly downscale the mesh to speed-up the rendering which is required to compute the object silhouette loss \mathcal{L}_{obj} . We compute watertight meshes for all ShapeNet models [9] using Manifold-Plus [21]. We rely on ACVD [62] to uniformly resample the object meshes, and set the target number of vertices to 1000 which we observe empirically to provide a satisfactory trade-off mesh approximation and rendering speed.

Collision loss. Unlike PHOSA [65] which penalizes collisions using a *local* loss [2] which acts at the mesh surfaces, we use an implementation which relies on Signed Distance Fields (SDF) [23]. We follow the original implementation, using a grid size of 32 for the signed distance field.

Local interaction heuristic. We use the loss term introduced by [19], which showed improved qualitative results when used in the context of fully-supervised learning. We accelerate computations by relying on the SDF computation from [23] to perform collision checking. We use the

same parameters as in the original implementation for the attractive and repulsive terms, 1cm and 2cm respectively.

Contact accuracy. We measure contact following [25], where contact is defined as negative hand vertex values in the object SDF.

Runtime. As we rely on learnt models for detection, segmentation and initial hand pose estimation, pre-processing 10 frames of dimension 640*480 takes less than a second on a Tesla V100 Nvidia GPU. The initial object fitting stage for 50 random object pose initialization takes around 10 second per frame. The joint fitting runs at 14 iterations per iteration in absence of collision and contact terms. With the fine-grained interaction constraints, the iteration time increases up to 4 iterations per second. In total, complete fitting of a 10 frame video clip takes between 2 and 3 minutes.