Computers & Graphics 116 (2023) 474–490

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Survey Paper

A survey of deep learning methods and datasets for hand pose estimation from hand-object interaction images[☆]

Taeyun Woo, Wonjung Park, Woohyun Jeong, Jinah Park [*]

Korea Advanced Institution of Science and Technology, Daejeon, Republic of Korea

ARTICLE INFO

Article history:
Received 12 April 2023
Received in revised form 28 July 2023
Accepted 25 September 2023
Available online 26 September 2023

Keywords:
Hand-object pose
Reconstruction
Computer vision
Benchmark dataset

ABSTRACT

The research topic of estimating hand pose from the images of hand-object interaction has the potential for replicating natural hand behavior in many practical applications of virtual reality and robotics. However, the intricacy of hand-object interaction combined with mutual occlusion, and the need for physical plausibility, brings many challenges to the problem. This paper provides a comprehensive survey of the state-of-the-art deep learning-based approaches for estimating hand pose (joint and shape) in the context of hand-object interaction. We discuss various deep learning-based approaches to image-based hand tracking, including hand joint and shape estimation. In addition, we review the hand-object interaction dataset benchmarks that are well-utilized in hand joint and shape estimation methods. Deep learning has emerged as a powerful technique for solving many problems including hand pose estimation. While we cover extensive research in the field, we discuss the remaining challenges leading to future research directions.

1. Introduction

Hand pose estimation is a promising research theme in VR/AR [1–4], human–computer interaction (HCI) [5,6], and robotics [7–12]. The goal of hand pose estimation is to understand the movement of a hand and to reproduce natural hand motions. Initial studies in the field adopted a marker-based capturing system, yet although such approaches can derive accurate hand poses, they are not suitable for widespread applications due to their complex setup and unavoidable post-processing. To overcome these inconveniences, many studies have explored markerless hand pose estimation from images that can be acquired by an ordinary camera, without any complex system. This image-based approach also allows the collection of more natural hand poses in action from everyday life recordings, compared to marker-based approaches, which may constrain the hand's movement.

Meanwhile, there have been many research works [13–16] proposed for hand pose estimation from images with remarkable results. However, most of these methods focus on estimating the poses of a hand alone and are thereby constrained to empty-hands scenarios, such as gesture recognition and sign language applications. Nonetheless, the primary role of the hand is to interact with its surroundings and the objects within them. Consequently, it is more appropriate to capture hand poses in hand-object interaction (HOI) scenarios for practical applications rather than empty hands alone. Existing approaches cannot infer accurate hand poses in HOI images due to the presence of objects in these images. Thus, the estimation of hand poses in HOI scenarios has become increasingly significant over time.

There are common challenges in hand pose estimation arising from HOI scenarios as well as from empty hands scenarios. Previously, image-based hand pose estimating techniques have struggled with self-similarity among fingers, self-occlusion, and high degrees of freedom in hand articulations. In addition to these challenges, capturing hands interacting with an object can involve mutual occlusion between the hand and the object. Also, the estimated hand pose has to be physically plausible; in other words, the hand and object cannot inter-penetrate each other.

Despite these aforementioned challenges, numerous methods have tried to capture hand poses from images for HOI scenarios. These techniques can be classified into three approaches: generative, discriminative, and hybrid. The generative approach optimizes hand models by minimizing their discrepancies with input images. In contrast, the discriminative method employs a statistical hand shape model constructed in a data-driven way. Each approach suffers from its own problems: the generative approach lacks the reality of a simple hand model and has high computational costs, while the discriminative approach faces a shortage of HOI datasets that contain plentiful varieties of grasping hands with objects and annotations. Meanwhile, hybrid approaches combine both generative and discriminative approaches

[☆] This article was recommended for publication by Dr. X. Han.

[*] Corresponding author.

E-mail addresses: taeyun.woo@kaist.ac.kr (T. Woo), fabiola@kaist.ac.kr (W. Park), whjeong@kaist.ac.kr (W. Jeong), jinahpark@kaist.ac.kr (J. Park).

https://doi.org/10.1016/j.cag.2023.09.013
0097-8493/© 2023 Elsevier Ltd. All rights reserved.

to exploit the benefits of both, but the problems remain in the hybrid approaches.

In recent years, the emergence of deep learning has led to outstanding techniques for hand pose estimation from HOI images. The reason for this remarkable performance is that neural networks can handle complex data in high dimensions, such as the manipulation of the hand. Furthermore, HOI datasets have appeared for training these neural networks, which generate more precise estimations in this field.

The objective of this survey is to provide a comprehensive review of both the state-of-the-art techniques for hand joint and shape estimation and the datasets of hand-object interaction images.

There are several surveys [17–20] that have examined hand pose estimation from images or videos. Oudah et al. [18] and Rastgoo et al. [20] reviewed the hand pose research, not considering hand-object interaction scenarios. Conversely, Huang et al. [19] provided a brief overview of hand tracking for HOI scenarios, but they mentioned only a few related papers. Ahmad et al. [17] examined various hand pose estimation and tracking methods for hand-object interaction scenarios, covering studies up to 2017. Unlike these surveys, we provide an extensive review of state-of-the-art techniques for hand pose estimation under interaction with an object. To review general hand pose estimation techniques, we recommend reading the survey from Ahmad et al. who ably describe several classical approaches. In contrast to these previously published surveys, our primary focus is to look into deep learning-based hand pose estimation approaches as well as the datasets that can be used to train such models. We anticipate that our survey paper will provide potential future directions to contribute to the further development of hand pose estimation in HOI scenarios.

In the following three sections, we introduce deep learning techniques for capturing hand poses from an image in which a hand interacts with an object: Section 2 illustrates research on estimating graph-like hand joints. Section 3 discusses studies about the reconstruction of 3D hand models in mesh or signed distance function (SDF) representations, which give *denser* than joints. And Section 4 introduces papers on the reconstruction of not only the hand but also 3D object models. Then we summarize available hand-object interaction(HOI) datasets in Section 5. Finally, Section 6 discusses the challenges and future direction of HOI research, and the conclusion follows in Section 7.

2. Hand joint estimation

The human hand's pose is commonly represented by annotating joint positions. The joint positions are connected by the skeletal configuration of the hand, and this skeletal chain provides a clear and simple depiction of the hand pose. Such joint representation is commonly used in both empty hands and hand-object pose estimation.

In this section, we review deep learning-based approaches that output the hand pose as a skeletal chain representation from an image of a hand interacting with an object. We divide the approaches into two subsections: (1) only on hand joint estimation, and (2) having both hand joints and object bounding box estimation. Table 1 shows the summary of hand joint estimation approaches, listed in chronological order, that we have covered in this section.

2.1. Hand pose estimation approaches in the form of a skeletal chain

Goudie and Galata [21] were the first to propose the use of an end-to-end deep-learning approach for hand joint estimation. They used a two-stage CNN network to estimate hand joints from a depth image. The first-stage CNN segments the image pixels into background, hand, and object. In the second stage, the segmented image from the first stage and the input depth image are concatenated and fed into the CNN which returns the 3D positions of 21 hand joints.

Choi et al. [22] utilized fused depth images, which combine both real and synthetic depth images, whereas previous methods used only real depth images. This approach was based on a technique that shows synthetic depth images can lead to better performance in hand joint estimation due to lower noise levels compared to real-world depth images [48]. The synthetic depth images can be obtained by an autoencoder from real-world depth image input. To represent both the realistic and synthetic domains, the generated image and the original image are fused together. From the fused image, two images are created with predicted center positions by a random forest classifier: the center of a hand and an object are at the center of each image. Then, the hand-centered image and object-centered image are fed into different networks separately. The two centered-images are fed into a grasp classification network, which predicts the global orientation and grasp type of a given image to reduce the search space among training images. The final hand pose is estimated by the nearest neighbor search from the restricted space that is similar to the input hand pose.

For real-time tracking of a hand-object interaction scenario, Mueller et al. [25] proposed a hand joint estimation method for RGB-D videos. For a given RGB-D input, they created a colored depth map that maps each pixel in the RGB channel onto the depth map to ignore camera-specific variations. The first CNN module regresses the 2D hand position heatmap. Then, the image that is cropped with respect to the hand position is fed into the second CNN module which predicts root-relative 2D joint heatmaps and 3D joint positions. Finally, the hand joint skeleton is optimized by 2D and 3D hand joint positions, and the temporal consistency between consecutive frames.

Baek et al. [33] suggested a domain adaptation method that maps the hand-object pose estimation to hand-only pose estimation. The hand-object datasets do not adequately represent the diverse aspects of grasping, and as such, they aimed to adapt the hand-object pose estimation to a more comprehensive benchmark area: hand pose estimation. The network takes an RGB image of a hand-object interaction as input and generates a corresponding hand-only image with joint position annotations. At the first stage, the network estimates spatial feature maps and heatmaps from the input image and generates a hand-only image, rendered from hand mesh using MANO [49]. Then, the feature maps and heatmaps extracted from the input image and rendered image respectively are fed into a GAN [35] to synthesize a refined hand-only image. Using the synthesized image, a hand mesh is generated again and finally hand joint positions are predicted.

Although previous approaches were run in a desktop setting, Yin et al. [43] presented a lightweight network aimed at achieving real-time hand-object pose estimation for mobile applications. Their proposed network builds on HOPE-Net [39], which is for both a hand and object pose estimation. However, they replaced the ResNet [26] encoder of HOPE-Net with MobileNet-v3 to quickly extract 2D hand key points from the input RGB image. Then, the 3D hand key points are predicted based on the 2D hand key points using Adaptive Graph U-Net [39].

Wen et al. [46] suggested a method that estimates both hand joint positions and hand-object action from an egocentric RGB video. Two estimations have different temporal granularity: hand pose estimating task is an instantaneous situation, but hand-object action is a time-lasting phenomenon. Hence, they estimated the hand pose to each frame, then the hand-object action is predicted by exploiting overall input frames. Firstly, ResNet-18 [26] extracts the image feature of each frame, and these

Table 1
A summary of hand joint estimation approaches. Under the loss function, we used a representative word to refer to its comparison with the ground truth and estimated one. (For example, 'Hand joint' means the loss is the difference between the labeled hand joint positions and their estimated positions.)

Method	Year	Input	Output	Model architecture	Loss function	Evaluation Metric[a]	Dataset(s)	Real-time
Goudie and Galata [21]	2017	Depth	Hand only	FCN	Hand joint	JE, Max error	Custom	X
Choi et al. [22]	2017	Depth	Hand only	Autoencoder FCN	Hand joint	CA, MJAE, MJE	NYU [23], GUN-71 [24]	X
Mueller et al. [25]	2017	RGB-D	Hand only	FCN ResNet [26]	Hand joint, Temporal consistency	2D PCK 3D PCK	EgoDexter [25]	O
Oberweger et al. [27]	2018	Depth	Both	STN [28] ResNet [26], FCN	Hand joint, Depth	MJE	NYU [23], Dexter+Object [29]	O (40 fps)
H+O [30]	2019	RGB video	Both	FCN LSTM [31]	Hand joint, Object pose, Confidence, Class.	CA, 3D PCK, 2D MJE	FPHA [32], EgoDexter [25]	X
Baek et al. [33]	2020	RGB	Hand only	CPM [34], GAN [35]	Hand joint, 2D Heatmap, GAN Loss	3D PCK, 2D PCK	Dexter-object [29], EgoDexter [25], SynthHands [25] HO-3D [36], STB [37], ObMan [38], RHD [13]	X
HOPE-Net [39]	2020	RGB	Both	ResNet-10 [26], GCN [40] AG U-Net [39]	Hand joint, Object pose	PCK, PCP	FPHA [32], HO-3D [36], ObMan [38]	O
Zhuang and Mu [41]	2021	RGB	Both	ResNet-50 [26], GCN [40]	Hand joint, Object keypoint, Physical affinity [41]	3D PCK, PCP	FPHA [32], ObMan [38]	X
Cheng et al. [42]	2021	2D pose	Hand only	FCN AG U-Net [39]	Hand joint	MPJPE, PCK	FPHA [32], HO-3D [36]	X
Yin et al. [43]	2021	RGB video	Hand only	MobileNet-v3 [44], GCN [40], AG U-Net [39]	Hand joint	Model size, GPU usage 2D PCK, 3D PCK	FPHA [32]	O (60 fps)
Zhang et al. [45]	2021	RGB	Both	CPM [34], GCN [40]	Hand joint, 2D heatmap	3D PCK, 3D PCP, AUC	FPHA [32], HO-3D [36], ObMan [38]	X
Wen et al. [46]	2023	RGB video	Hand joint, hand-object action	ResNet [26], FCN	Hand joint, object classification action classification	PCK, AUC, JE, CA	FPHA [32], H2O [47]	X

[a] The abbreviations of evaluation metrics are summarized in Appendix.

features are grouped in several groups with other neighboring frames. These groups are fed into each pose estimating block, and the block predicts both hand pose and object labels. Finally, the action estimating module inputs all estimated hand poses and object labels from all groups, and predicts the hand-object action.

2.2. A hand joint with an object estimation

Estimating a hand pose only in a hand-object interaction scene is neither natural nor straightforward. When a hand is interacting with an object, it makes more sense to estimate the hand pose considering the object in hand. While the hand pose is depicted as a skeletal chain, the object's pose is represented in a cuboid bounding box with eight vertices.

Oberweger et al. [27] presented a deep learning-based approach to enhance the conventional hand-object pose estimation process. Their method involves a feedback loop consisting of a localizer, pose predictor, image synthesizer, and pose updater. Firstly, the localizer crops the input image of the target (hand-object) and centers them using the Spatial Transformer Network [28]. The pose predictor then estimates the pose of the cropped image, while the synthesizer generates a synthesized image from the pose. The updater utilizes both the original image and the synthesized image to estimate the pose variation and update the pose accordingly. This loop iterates multiple times to produce a more accurate pose. The hand and object loops are separate, and the hand pose estimation loop involves predicting the hand joint by using the hand pose predictor and producing an image with the hand image synthesizer. Likewise, the object pose predictor predicts an object bounding box, and the object image synthesizer generates an object image.

While the prior approach utilized a single-depth image, Tekin et al. [30] introduced a method for hand-object pose estimation from RGB video. It can also recognize the action of hand-object interaction. Firstly, a fully-convolutional network (FCN) divides the input frame of the video into a 3-dimensional grid, spanning the 3D scene in front of the camera viewpoint. The 3D grid consists of cells that have a discretized size and contain both hand pose information and object pose information in each cell. Each piece of information contains keypoint positions, the action of hand-object interaction, and a confidence value. To ensure accurate pose estimation, they employed a confidence value that scores higher when a hand or object is visible in the cell. The hand pose and object pose are regressed from the cells that have high confidence value. In addition, these cells that have a higher confidence score are fed into the RNN, which recognizes hand-object interaction actions considering current and previous frames for temporal consistency.

While the previous approaches regress the joint positions through the fully-connected layer, these three methods utilized GCN. As the FC layer is incapable of modeling the connectivity among keypoints and can only regress their positions, a more appropriate approach would be to use GCN, a specialized network that can explicitly handle graph structures like the hand joint chain.

Doosti et al. [39] proposed a hand and object pose estimation model HOPE-Net, that employs Adaptive Graph U-Net that converts 2D joint positions to 3D positions. From an RGB image input, ResNet-10 [26] extracts the feature, and the 2D keypoints of the hand are roughly predicted. Based on the predictions and the extracted feature, the initial adjacency matrix for a graph convolution is constructed. Then, the adaptive graph U-Net substitutes a 2D graph for a 3D graph through fully-connected layers to pool and unpool the graph. In addition, they eliminated the node-removing process of the conventional graph U-Net [50] since each node of a hand skeleton has its own role.

Cheng et al. [42] introduced a semi-supervised 3D hand-object pose estimation network to given 2D pose of a hand-object with an object-oriented coordinate system, where the joints' positions are invariant to the camera perspective. They designed a two-phase procedure to train the pose estimation network using both labeled and unlabeled data. In the first phase, they trained the dictionary learning module using labeled data through a self-reconstruction process. The reconstruction reproduces the hand-object pose in camera-coordinates for the object-oriented coordinates. Then, the pose estimation module is trained on both the labeled and unlabeled data. The module is based on Adaptive Graph U-Net [39] which takes a 2D pose and generates a 3D pose in a camera coordinate. The 3D pose is fed into the freeze dictionary learning module to transform into the object-oriented 3D pose. At the inference time, only the pose estimation module is used to generate a 3D pose. The proposed network has the potential for various hand-object interaction applications where labeled data is limited.

Similar to HOPE-Net, Zhuang and Mu [41] designed the hand-object pose estimation pipeline with GCN [40], with a new physical affinity loss that considers the grasping stability based on kinematics. From an RGB image input, ResNet-50 [26] extracts a feature and predicts the 2D hand pose and 2D object pose. With the initial poses of the hand and object, graphs are constructed by concatenating the 2D coordinates and the extracted feature. Through the GCN layers, the initial graphs are refined and 3D poses of the hand and object are obtained. Even though this kind of *coarse-to-fine* GCN network has been used at HOPE-Net with Adaptive Graph U-Net, it did not treat all nodes of both a hand and object graph together. Instead, the authors employed two sub-graph networks to capture each aspect of a hand and object, which are subsequently merged into a single graph. To reflect the physical constraints of the contact between the hand and object, a physical affinity loss is introduced. First, the contact points on the object bounding box's faces are sampled that contact with hand joints. Then, the loss is calculated as a sum of cosine similarity among vectors that start from an object center to each contact point.

While the two approaches [39,41] suggested a GCN-based network, Zhang et al. [45] suggested an interaction-aware approach between the hand graph and object graph. The given RGB image input is first used to estimate 2D heatmaps that localize 2D keypoints based on the Convolutional Pose Machines (CPM) [34]. The obtained 2D heatmaps and extracted features from CPMs convert to 3D poses of a hand and object similar to the work by Zimmermann et al. [13]. When the initial poses are obtained, they are fed into InterGCN that refines the poses considering features from the interaction between hand and object. To utilize the GCN manner, two kinds of graphs are introduced: general relation graph (A^g) and interaction-specific relation graph (A^s). The A^g's adjacency matrix is constructed based on the findings that keypoints have strong relations when they are close to each other. Thus, if the distance between a hand keypoint and an object keypoint are close, they become a neighbor. Furthermore, not only the near keypoints affect the hand-object interaction. Hence, the adjacency matrix of A^s is constructed by an attention-mechanism [51] to exploit long-range interaction among far keypoints. In conclusion, this InterGCN outputs the refined poses of the hand and object, considering the context of hand-object interaction.

3. Hand shape reconstruction

The hand joints estimation methods discussed in Section 2 provide valuable insights into hand-object interaction, particularly in terms of how the human hand grasps an object and where it makes contact. However, relying solely on joint positions provides only limited *sparse* information of the hand pose and contact region on the object's surface. For these reasons, in this section, we discuss reconstruction approaches that generate the hand 3D model from image-based input which can give *denser* annotations rather than hand joint positions. We introduce approaches that reconstruct the hand shape, however, they do not create the object shape. Instead, some methods estimate the 6D pose of the given object mesh as input. Furthermore, to utilize the hand-object interaction, we bring the research work that introduces feature fusion: reconstructing a hand shape with fused features of both the hand and an object. The summary of Section 3 is listed in Table 2.

3.1. MANO

To represent a complex hand manipulation, most approaches have utilized MANO [49]. MANO stands for a hand model with articulated and non-rigid deformations, and a parameterized hand model developed by Romero et al. They collected around 1000 instances of hand scans of 31 subjects, and extracted the principal components by PCA. Based on principal components, the hand mesh can be reconstructed simply with MANO parameters. Typically, to obtain a hand model by MANO, previous approaches have utilized two kinds of MANO parameters: pose and shape parameters. The pose parameters imply the movement of each joint based on hand kinematics, and the shape parameters represent the hand's appearance.

In terms of deep learning, Hasson et al. [38] designed a differential MANO layer that estimates the MANO pose and shape parameters from the encoded image features. Thanks to their work, many hand-object studies adopted the differential MANO layer into their network architecture, and it facilitates to utilize hand's prior knowledge within MANO. We suggest referring to Romero et al. [49] and Hasson et al. [38] for more in-depth information about MANO and the differential MANO layer, respectively.

3.2. Hand shape reconstruction without object

Shan et al. [52] proposed an approach to reconstruct the hand mesh from everyday hand-object interactions on YouTube. Firstly, given a single image of the video, they crop the hand and object using Faster-RCNN [53]. With the cropped image, they inferred hand sides (left and right) and contact state of the hand and object. Then, they reconstructed the MANO hand model from the cropped hand image through a differential MANO layer [38] that estimates MANO parameters. Their training dataset has annotations of the bounding boxes and contact of hand and object, however, does not have the ground truth of a hand mesh. Thus, they checked whether the estimated hand mesh is plausible or not by comparing joint positions regressed from the estimated mesh as in [79]. Not only can their method reconstruct hand models from a hand-object image, but it can also estimate the hand model from an object-only image.

Unlike the former methods that directly reconstruct a hand mesh from an extracted image feature, Park et al. [80] proposed a HandOccNet that reconstructs a hand mesh through attention modules. Their approach is based on the findings that the occlusion by an object can provide highly rich information about a hand. The necessity map, representing where hand exists, is estimated from the extracted features through the ResNet-50 [26]-based FPN [70]. The feature and necessity maps are used to make a primary feature (F_P) and a secondary feature (F_S) that represents the hand information and non-hand information, respectively. Then, the feature injecting transformer (FIT), takes F_P and F_S and injects the knowledge of F_P into F_S considering the correlation. Usually, Transformer [51] is based on the softmax-based attention mechanism. However, the authors also used a sigmoid-based attention mechanism to prevent undesired high-attention scores between unrelated pixels, such as between background

Table 2
A summary of hand mesh reconstruction approaches from a given image and (optional) given object mesh. Under the loss function, we used a representative word to refer to its comparison with the ground truth and estimated one. (For example, 'Hand joint' means the loss is the difference between the labeled hand joint positions and their estimated positions.)

Method	Year	Input	Model architecture	Loss function	Evaluation Metric[a]	Dataset(s)	Real-time
Shan et al. [52]	2020	RGB	ResNet-101 [26], Faster-RCNN [53] Diff. MANO layer [38]	Hand grasp class, RoI of Hand, RoI of Object	Average precision	VLOG [54], VGGHands [55], 100DOH [52], VIVA [56], EgoHands [57], TV+Co [58]	X
Huang et al. [59]	2020	RGB	ResNet-50 [26], GCN [40], NART [59]	Hand joint, Obj pose, Anatomical Loss, MANO param.	Hand MME, PCK-AUC, Obj B.B MME, PCP-AUC	FPHA [32], HO-3D [36], ObMan [38]	O
Hasson et al. [60]	2020	RGB video	ResNet-18 [26] Diff. MANO layer [38]	Hand mesh, Hand joint Obj mesh, MANO param, Temporal consistency	Hand MME, PCK, Obj B.B MME	FPHA [32], HO-3D [36]	X
Hasson et al. [61]	2021	RGB video	PointRend [62], FrankMoCap [63]	Hand mesh, MANO param, Obj mask, Scale, Smooth, Collision, Hand-obj dist.	MME, PD, ADD-S, F-Score, CR	HO-3D [36], CORe50 [64], EPIC-KITCHENS [65]	X
Cao et al. [66]	2021	RGB	PointRend [62], FrankMoCap [63]	MANO param, Obj mask, Obj depth, Collision, Hand-Obj dist.	MJE, CD	FPHA [32], HO-3D [36]	X
Yang et al. [67]	2021	RGB	ResNet-18 [26] PointNet Encoder [68]	Vertex contact, Contact region, Elasticity, Anatomical term	MME, PD, IV, DD, SD	FPHA [32], HO-3D [36]	X
Liu et al. [69]	2021	RGB video	ResNet-50 [26], FPN [70]	Hand mesh, Obj pose, Hand joint heatmap, confidence	MJE, MME, F-Score, PCK-AUC, PCV-AUC, ADD-0.1D	HO-3D [36], FPHA [32] , sth-sth [71]	X
Hampali et al. [16]	2022	RGB	U-Net [72], self-attention [51], cross-attention [51]	Hand joint heatmap, Obj keypoint heatmap Hand joint, Obj pose, MANO params, translation	MPJPE, MRRRPE, MSSD, PCK-AUC,	InterHand2.6M [73], HO-3D [36], H_2O-3D [16]	X
Wang et al. [74]	2023	RGB	ResNet-50 [26], Moon et al. [75], GCN [40]	Hand joint, Obj silhouette, Hand mesh, Obj pose	MJE, MME, PD, CP	DexYCB [76], HO-3D [36]	O (34 fps)
Fu et al. [77]	2023	RGB video	ResNet-50 [26], transformer [51], GAN [35]	Hand mesh, adversarial loss, auxiliary loss [78]	MPJPE, AUC, F-score	DexYCB [76], HO-3D [36]	X

[a] The abbreviations of evaluation metrics are summarized in Appendix.

and hand pixels. After the FIT outputs an attention map with the two attention mechanisms, the attention map is enhanced through the self-enhancing transformer to estimate MANO hand model.

Fu et al. [77] concentrated on both spatial and temporal continuity of the hand from input image sequences. Their model is sequentially organized with spatial transformers [51], temporal transformers, and dynamic fusion modules. They utilized the transformer due to its ability to deal with long-range dependency, while the CNN contains a local inductive bias that is vulnerable to occlusion. Based on this character, firstly, the spatial transformer encodes each frame through ResNet-50 [26] with positional embedding to given input image frames. The encoded latent features of all frames are fed into the temporal transformer that enhances the hand feature by exploiting the correlation of the hand in every frame. Then, the dynamic fusion module estimates a hand pose in the current frame, utilizing information from neighboring frames explicitly in a confidence-driven manner. Finally, the estimated hand pose of each sequence passes through the motion discriminator based on GAN [35].

3.3. Hand shape reconstruction with object pose

Hasson et al. [60] provided a method to reconstruct the hand mesh from an RGB video using novel photometric loss in semi-supervised training. The photometric loss is used to regulate the inconsistency between an annotated frame and an unannotated frame. It calculates the 3D displacement of two frames (optical flow) and the silhouette of the hand-object model. To estimate a hand mesh, the network regresses MANO [49] parameters. Unlike the hand mesh, the object mesh is already given. Thus, this approach estimates a 6D pose in the form of a rigid transformation. Their method reconstructs the hand-object model to the given RGB video, and utilizes on a sparsely supervised learning approach that encourages consistency via photometric loss.

In addition, Hasson et al. [61] proposed a hand-object reconstruction method via an optimization-based fitting approach from RGB video. 3D annotations (*i.e.* meshes, 3D joint positions) are not always available and are hard to get. Due to a lack of 3D annotations, they proposed to reconstruct the hand-object without any 3D supervision. Instead, they utilized segmentation maps and hand joint positions. First, an off-the-shelf object detection and instance segmentation approach [62] predicts hand and object pixels. Then, to each frame, FrankMocap [63] estimates an initial hand mesh at the pixel level and camera intrinsic if it is not given. The segmented object mask is used to initialize the object pose for a given object mesh, minimizing the differences between the observed frame and rendered object model. Finally, the hand and object meshes are refined jointly through the optimization process.

While Hasson et al. trained on an in-lab setting data, Cao et al. [66] proposed a hand reconstruction method from an RGB image that is captured from everyday life. First, the MANO hand mesh with parameters and camera intrinsic were estimated by FrankMocap [63]. Note that their training dataset does not contain a ground truth of hand mesh. Thus, they verified the estimated hand mesh by comparing projected 3D hand joint positions, regressed from the hand mesh and the ground truth 2D hand joint positions. Then, an object transformation is predicted, and a differentiable renderer [81] generates a 2D segmentation map and depth map to compare with the input image. When the hand and object mesh are acquired, the two meshes are adjusted jointly considering the distance between a hand and object and the inter-penetration between them. However, the above losses are not sufficient for a physically-plausible model in terms of the contact area. Hence a pose refinement network is used, trained with GRAB [82] that contains contact regions for objects. This improves the hand mesh and object mesh regarding the contact region between them.

Huang et al. [59] adopted a non-auto regressive transformer (NART) [83] to generate a 3D hand-object pose and hand mesh. 2D keypoint positions and the 3D hand-object reference pose are predicted using ResNet [26] from an input image. With the extracted feature from ResNet and reference pose, the NART decoder provides a refined 3D hand-object pose. Given the extracted features and estimated 3D hand-object pose, a hand mesh can be generated by MANO [49].

Hampali et al. [16] suggested a method to localize keypoints and recognize corresponding hand joints from an image. Although their work mainly focuses on the pose estimation of two hands, we introduce their work since they also demonstrated hand mesh reconstruction. Their localizing stage identifies relevant sites (keypoints) with the hand or object through U-Net [72] in the form of a segmentation map. Then, the segmentation maps and extracted features from a U-Net are encoded to predict candidates of the hand and object keypoints. For hand pose estimation, the predicted hand keypoints are fed into a self-attention [51] module for filtering the hand keypoints. Then, a cross-attention [51] layer predicts which keypoint corresponds to each joint of the hand. From the hand joints, the hand mesh can be reconstructed with MANO parameters. In the case of object pose estimation, both the hand and object keypoint candidates are fed into the self-attention module and the final object pose is predicted with the cross-attention layer similar to the hand.

For a physically-plausible hand-object model, some approaches dealt with the inter-penetration between a hand and object mesh implicitly via their loss functions. However, Yang et al. [67] designed the contact states explicitly called Contact Potential Field (CPF). The CPF is built on a spring-mass system, and it refines an estimated hand mesh with an attractive and repulsive energy term between the hand mesh and the given object mesh. They also proposed an anatomically constrained MANO hand model called A-MANO. It restricts the rotation axes of each finger and contains anchors in charge of computing contact states like spheres on hand models as in [84,85]. Firstly, the hand mesh parameters and object 6D pose are estimated by [60]. The parameters generate the hand mesh and the given object mesh is aligned with the estimated pose. Afterward, contact regions of the object mesh and corresponding anchors on the A-MANO are predicted. In the end, an iterative optimization stage refines the hand-object model to reflect contact regions and anatomical constraints via A-MANO.

By incorporating the contact region into hand pose estimation, we can achieve a more realistic hand pose. For further insights into this topic, we recommend referring to the works of Grady et al. [86] and Jiang et al. [87]. These researchers have proposed approaches to estimate contact regions on given meshes.

3.4. Hand shape reconstruction with feature fusion

The approaches described in the previous sections reconstruct the hand mesh directly by estimating MANO parameters from the extracted hand feature. However, they are not natural in terms of hand-object interaction. The hand and object shape themselves give key insights into each other: the hand pose is highly correlated with the object shape and features, and vice versa. For this reason, feature fusion between a hand and object feature (which passes through the encoder) has been invented, and those fusion-based approaches are described in this section.

Liu et al. [69] introduced an approach for 3D hand mesh reconstruction with feature fusion. Also, they suggested a semi-supervised learning pipeline with their method. First, they extracted features of the hand and object using ResNet-50 [26]-based FPN [70] in separate branches. The object feature is updated, fusing the hand feature based on an attention mechanism [51]. Then, a hand decoder generates 2D hand joint positions and MANO parameters, and an object decoder predicts the 6D pose of an object. For the semi-supervised learning pipeline, their estimating module generates pseudo-hand labels from unlabeled data. Incorrect pseudo-labels are filtered by spatial consistency constraints and temporal consistency constraints. The spatial consistency term includes hand bounding boxes, the joint distance between the label and estimation, and physical-plausibility of the hand. Due to taking the video input, they utilized the optical flow to derive the temporal consistency. Finally, the network is retrained in semi-supervised manner with the union of the fully annotated dataset and a high-confident pseudo-labeled dataset. This pipeline helps the network to improve the estimation results and generalization, exploiting the diversity in unlabeled dataset from daily situations with estimated pseudo-labels.

Although the former method fused the feature of the hand and object through their attention mechanism, it did not consider the spatial information on the mesh during the feature fusion stage. To address this issue, Wang et al. [74] suggested a technique that can consider the dependency between hand and object shape via their dense mutual attention mechanism. First, ResNet-50 [26] is used to extract image features, and the initial meshes of the hand and object are obtained, following the method of Moon et al. [75]. For a physically-plausible hand-object model, the initial hand mesh is refined through GCN and their mutual attention mechanism. From the initial meshes, the hand and object graphs are constructed. Then, the GCN layers update each node feature, exploiting surrounding node features. Finally, mutual attention layers refine the hand mesh and estimate the object pose by calculating the attention map between the hand and object graph nodes to combine each feature.

4. Hand and object shape reconstruction

In Section 3, we described two types of approaches: those that focus on reconstructing only the hand shape and those that use given (known) object models to estimate the hand shape and object pose. However, the latter approach can only be applied in scenarios where an object model is available, and therefore, is not suitable for object-agnostic situations. In this section, we

Table 3
A summary of reconstruction approaches of both a hand and object mesh together. FCN means the fully convolutional network that is designed by the authors of the method. Input is basically the 'image' unless indicated otherwise.

Method	Year	Input	Model architecture	Loss function	Evaluation Metric[a]	Dataset(s)	Real-time
Hasson et al. [38]	2019	RGB	ResNet-18 [26] Diff. MANO layer [38] AtlasNet [89]	Hand joint, Hand mesh, Mano params, Obj mesh, Contact	MME, PD, IV, SD	ObMan [38], HO-3D [36] CORe50 [64], FPHA [32]	X
Zhang et al. [90]	2019	Two depthvideo	DenseSegAttention [91] LSTM [31]	(opt) hand, object, collision, optical flow	Pixel error	–	O (25 fps)
Grasping Field [92]	2020	RGB	ResNet-18 [26] FCN	Hand mesh, Obj mesh, Contact, Penetration	IV, MME, SD, CP, perceptual score	ObMan [38], FPHA [32], HO-3D [36]	X
Chen et al. [93]	2021	RGB	ResNet-18 [26] Diff. MANO layer [38] AtlasNet [89], LSTM [31]	Hand joint, Hand mesh, MANO params, Obj mesh, Depth, Contact	Hand MME, PCK, MPJPE, CD, PD	ObMan [38], FPHA [32]	X
Almadani et al. [94]	2021	RGB-D,TSDF	Adaptive Graph U-Net [39] GCN [40]	hand, object	Hand MME, Obj MME	HO-3D [36], FPHA [32]	X
Zhang et al. [95]	2021	depth video	ResNet [26] (s/o) DetNet [96]	(opt) hand, object, contact, optical flow	PCK, MIoU, MME	–	O (25 fps)
Tse et al. [97]	2022	RGB	ResNet-18 [26] Differential MANO layer [38] AtlasNet [89], GCN	Hand joint, Hand mesh, Obj mesh, Associative	MME, PD, IV F-score	ObMan [38], FPHA [32] , DexYCB [76], HO-3D [36]	X
AlignSDF [98]	2022	RGB	ResNet-18 [26] DeepSDF decoder [88]	Hand joint, MANO params, Obj mesh, Obj pose	MME, CP, PD, IV	ObMan [38], DexYCB [76]	X
Ye et al. [99]	2022	RGB	FrankMoCap [63], ResNet [26], DeepSDF decoder [88]	Hand SDF, Obj SDF, Contact	CD, IV, F-Score	ObMan [38], HO-3D [36] MOW [66]	X
THOR-NET [100]	2023	RGB	ResNet-50 [26], FPN GraFormer [101]	Hand joint, Obj pose Heatmap, Hand mesh, Obj mesh, Class Optical flow	PCK, MME, MPJPE	HO-3D [36], H2O [47]	X
Chen et al. [102]	2023	RGB videos	ResNet-18 [26], FCN	Hand joint, ordinal loss [103] obj pose, SDF	CD, JE, F-score	ObMan [38], DexYCB [76]	X

[a] The abbreviations of evaluation metrics are summarized in Appendix.

provide a comprehensive review of the methods that are capable of reconstructing both hand and object shapes from images. They utilize different models, other than the MANO [49] hand model, such as a Graph Convolution Network (GCN) [40] and the signed distance function with DeepSDF [88]. We categorize the methods based on how the hand and object mesh are generated. We also describe real-time approaches at the end of this section. The summary of the methods in Section 4 is shown in Table 3.

4.1. MANO hand model and AtlasNet

Hasson et al. [38] proposed a method that generates both hand and object meshes from a single RGB image. ResNet-18 [26] was used to extract hand and object features from an image, and each feature is fed into the hand branch and object branch respectively. At the hand branch, MANO [49] parameters are estimated in the differentiable MANO layer using the extracted hand feature. At that time, the AtlasNet [89] reconstructs the normalized object mesh from the extracted feature in the object branch. After the normalized object mesh is constructed, it is adjusted with translation and scale parameters from the hand branch to fit the hand mesh and object mesh jointly. Finally, to achieve a physically-plausible hand-object mesh model, they employed penetration and contact information within training loss.

Chen et al. [93] suggested a reconstructing approach similar to the previous method [38]. In addition to Hasson's approach [38], they utilized a depth estimation module and a feature fusion technique. The depth estimation module predicts the depth channel from RGB input. The feature fusion combines the hand feature and object feature from each branch, utilizing long short-term memory (LSTM) [31]. Their feature fusion method is based on an observation that the hand contains richer prior knowledge than the object because the hand can be represented in MANO parameters. Through the LSTM blocks, the object feature is enhanced, exploiting the hand feature. This enhancement in object feature helps reduce the chamfer distance between the ground truth object mesh and the reconstructed mesh. In addition, the depth estimation module improves the accuracy of a predicted hand shape.

Also, Corona et al. [104] proposed a GanHand to reconstruct the hand grasp to a given objects-only image. The GanHand can be used not only when the object mesh is given (pose estimation), but also when the mesh is not provided. In the context of object mesh reconstruction, they used AtlasNet [89] to reconstruct a single object within the single object scenario. If there are multiple objects, one object is randomly selected, and its object mesh is reconstructed. With the silhouette from the reconstructed object mesh and the input image, the MANO parameters for a hand mesh are estimated. However, the parameters are not used to generate the hand mesh, instead, they were used to refine a template hand mesh. In GanHand, the initial template hand mesh is generated via a classification problem: given input image, the network predicts a class of grasp among grasp taxonomies [105]. From the estimated hand class, a hand template model is designated. Then, the refinement layer takes the template hand and predicts MANO parameters, and outputs deformation to refine the template hand model.

4.2. GCN

Previously, a hand mesh was generated from estimated MANO [49] parameters, and AtlasNet was adopted to predict the object mesh. Although AtlasNet is able to produce fine-quality meshes, it is not suitable for refining meshes to alleviate the inter-penetration between the hand and object mesh. Accordingly, recent studies [94,97,100] attempted to incorporate the Graph Convolution Network (GCN) [40] that effectively processes graph-based data like meshes.

Almadani et al. [94] proposed an approach for reconstructing both a hand and object from an RGB-D image and voxelized RGB-D in the form of a truncated signed distance function (TSDF). The TSDF of RGB-D image is generated by [106], and each RGB-D image and TSDF pass through different encoders. Then, the shape reconstruction process is two-fold with a concatenated feature. At the first stage, the initial 2D hand joint and object pose are estimated, and then converted to the 3D pose of hand-object by Adaptive Graph U-Net [39]. Then, the estimated 3D poses are used to reconstruct the hand-object mesh through sequential GCN modules.

While the former approach needs depth information, Tse et al. [97] reconstructed a hand-object mesh from an RGB image using GCN. They tried to jointly reconstruct a hand and object, fusing a hand feature and object feature like Chen et al. [93]. Their method is based on the following two observations: (1) estimating a hand-object shape is a highly-correlated with each other and (2) the occlusion can give meaningful information to both the hand and object. First, they extracted each feature of a hand and object by ResNet-18 [26], and an initial hand mesh and initial object mesh are generated by the differential MANO layer [38] and AtlasNet [89], respectively. Then, each hand and object initial mesh are refined with attention-guided graph convolution. It aggregates neighboring nodes in the mesh, and outputs the feature of each graph. For collaborating between a hand and object, each feature feeds into other branches' encoder: the extracted features from the hand graph and image feed into the object encoder and vice versa. Also, an unsupervised associative loss leverages feature transfer among similar object classes, and is applied during the training procedure.

Aboukahadra [100] suggested a method for two hands and an object reconstruction utilizing Graformer [101] instead of the attention-module [51] with GCN. Graformer, a Graph convolution transformer for 3D pose estimation, is adopted to estimate the hand and object pose, and reconstruct their shapes. First, the 2D heatmaps that represent keypoints of the hand and object are predicted by the Keypoint RCNN, which the authors of the Mask-RCNN [107] created to estimate the heatmaps of keypoints. As Almadani et al. [94] suggested a *corase-to-fine* strategy, GraFormer reconstructs the hands-object shape from the 2D graph. The initial 2D graph consists of 29 nodes (21 hand joints and 8 vertices of an object bounding box) and each node contains 2D heatmaps and feature maps from the middle of the Keypoint RCNN. The hands and object meshes are created by applying the Quadric Edge Collapse Decimation algorithm (QECD) [108] from a simple sphere mesh. During the training stage, the model employs the discrepancy loss between the ground truth and predicted outputs for both the hands and the object. Additionally, a photometric loss is also used to leverage the consistency similar to Hasson et al. [60].

4.3. Signed Distance Function (SDF) as intermediate representation

Two methods [92,98] adopted SDF which can represent volumetric information as an implicit surface function. The advantage of SDF is that it is easy to model contact regions and interpenetration. These aspects are important for reconstructing a physically-plausible hand-object mesh.

Karunratanakul et al. [92] proposed Grasping Field, that utilizes SDF to represent the shape of a hand and object. The proposed method can accomplish two tasks: hand grasp generation *given* 3D object, and hand-object model reconstruction from an RGB image. We concentrate on the second task. The input of the network is an RGB image and query 3D points, then the network outputs signed distance field to the hand and object surface separately on every query point. In addition, the network classifies a corresponding hand part of query points with zero signed distance from the hand surface. During training, the penetration loss and contact loss encourage the construction of a physically-plausible hand-object model, and classification loss is used for the classification of hand parts. Finally, hand and object meshes are reconstructed from each signed distance function through Marching cubes [109].

Even if the Grasping Field [92] achieved remarkable performance with SDF, there remain two limitations: (1) they did not adopt explicit prior knowledge, such as MANO [49], which benefits reconstruction of the hand shape; and (2) they did not disentangle a shape and 6D pose. For these reasons, Chen et al. [98] proposed AlignSDF, which incorporates prior knowledge and disentangles the shape and pose of each mesh. The inputs of AlignSDF are the same as the Grasping Field (an RGB image and query point), however, they utilized MANO parameters to inject prior knowledge. The hand and the object 6D pose are estimated separately. Note that the object is predicted in a wrist-oriented coordinate, so only the translation vector is estimated for object pose. To estimate the shape of a hand and object, each network decoder outputs a hand SDF and object SDF, respectively. They did not facilitate the loss that explicitly expresses inter-penetration like Grasping Field. Instead, they implicitly regularized the penetration based on a comparison with the ground truth which is already physically-plausible. Finally, the hand-object mesh is generated by the Marching cubes.

Ye et al. [99] proposed a method that utilizes SDF to reconstruct objects with FrankMocap [63]. They aimed to reconstruct a hand-held object given in an image, based on the following insight: a hand manipulation provides a cue to predict the object shape within the hand. At the first stage, FrankMocap estimates the hand articulation and camera intrinsics. The hand mesh is reconstructed on the estimated hand articulation by rigging the MANO model. Then, the positional embedder encodes the hand articulation to articulation-aware encoding to predict the object's shape. Simultaneously, the ResNet [26]-based FPN visual encoder extracts features at different resolutions. Finally, with cascaded image features from the visual encoder and articulation-aware embedding, the decoder based on DeepSDF [88] outputs the SDF along the object surface.

The SDF can estimate the shape of the hand and object regardless of resolutions. However, it cannot explicitly estimate the shape based on underlying geometry, including the kinematics of the hand and manipulation of the object. In this sense, Chen et al. [102] suggested geometry-driven SDF (gSDF) that can reconstruct hand-object shapes from RGB images with geometry. First, the kinematic features of the hand and object are extracted in different branches. Especially, hand's kinematic features are obtained using inverse kinematics from estimated hand joints. In contrast, the object's kinematic features are defined with respect to the object's center. In addition, they adopted spatial–temporal transformer [110,111] to correlate neighboring image frames, and this strategy can enhance the local visual feature of input images. Finally, the hand SDF and object SDF are reconstructed from each feature and the local visual feature.

4.4. Hand-Object reconstruction operated in real-time

Some studies have aimed to reduce inference time to be utilized in real-time applications such as VR/AR. Zhang et al. [90] proposed a method that reconstructs hand and object meshes in real-time from two depth videos captured from opposite camera viewpoints. The hand and the object are segmented by DenseAttentionSeg [91] and the hand pose is predicted by LSTM [31]. Then, the hand pose and the segmented object are optimized using the hand energy term, object energy term, and interaction constraint term. The first two terms are for preserving consistency with the input, and the last term is to follow physical-plausibility. The hand mesh is reconstructed with sphere-mesh [112] hand model and the object mesh is reconstructed using DynamicFusion [113].

They also developed a new method [95] that uses a single depth camera to reconstruct hand and object meshes. Instead of estimating a hand-object segmentation map and hand pose separately, they used the idea of DetNet [96] to perform both tasks simultaneously. First, feature maps extracted from an input depth image by ResNet [26] are used to make heatmap and depth map of hand keypoints. Then, the heatmap and depth map cooperate with camera parameters to obtain 3D hand joints and goes into decoders to produce a hand-object segmentation map. Finally, the hand and object mesh are reconstructed with the same manner proposed in their previous work [90].

Both works [90,95] succeeded in running at 25 fps. However, they cannot reconstruct invisible parts of objects, ambiguous objects, and highly-occluded objects.

5. Datasets

The deep learning approaches have achieved state-of-the-art performance in estimating hand pose in scenarios where hands are interacting with objects. This achievement is largely attributed to the availability of training datasets that cover various aspects of hand-object interactions. In this section, we provide a thorough summary of existing datasets that captured hand-object interactions including their annotations, properties, and capturing methods. Table 4 is a list of datasets for hand pose estimation research. We marked some of the popular datasets (**bold** text in the table) for hand-object interaction. In addition, we describe the online data generation module [114] which can be adapted to an arbitrary hand-object estimation framework. Finally, we suggest a brief strategy to select proper datasets with respect to the target tasks at the end of this section.

5.1. Real-world datasets

A training dataset decides what domain a training network learns about. The objective of this research field is to accurately estimate hand pose from a given observed image for various application scenarios. Many researchers have captured human hands and objects in real-world settings using a markerless motion capture setup to construct real datasets that accurately reflect hand-object interactions. This section outlines the properties of datasets, labeling methods, and setup systems as well as their annotations.

5.1.1. FPHA

FPHA [32] is a First-Person Hand Action benchmark. The dataset contains 1,175 action videos belonging to 45 action categories in an RGB-D image modality. The 105K RGB-D frames were annotated with 3D hand joints and action classes. 3D hand joint positions were taken from six magnetic sensors on the hand and those positions are used to infer the hand joints' kinematic chain via inverse kinematics. The annotations of objects are 6D pose, which includes 3D location and orientation. The four kinds of object meshes are given in 10 different action categories. The six subjects participated and they are all right-handed.

The capturing RGB-D camera (Intel RealSense SR300) is attached to the shoulder of the participant. The six magnetic sensors represent five fingertips and one wrist, and the inverse kinematic approach, same with [131], reconstructs 21 joint positions of the hand. Note that the magnetic sensors are white, so they appear in RGB HOI images. On the other hand, their small size (2 mm) is negligible in depth data. The magnetic sensor is also attached to the nearest point from a center of the object to get the object's pose.

5.1.2. EPIC-KITCHENS

The EPIC-KITCHENS [65,127] dataset is an egocentric video dataset benchmark, especially in kitchen scenarios. There are two versions: EPIC-KITCHENS-55 [65] and EPIC-KITCHENS-100 [127].

In EPIC-KITCHENS-55, there are 11.5M frames with annotations of actions and object bounding boxes. In the recording process, 32 participants wore a helmet on which the camera (GoPro Hero7 Black) is mounted and the person does something in their kitchen. After the recording is finished, the participants annotated the name of the action themselves, such as "hold a mug". The recording is started as soon as they enter the kitchen and is stopped when they leave the kitchen. With narrations by the participants, manual transcripts were collected via Amazon Mechanical Turk (AMT). Likewise, the object bounding boxes are gathered with AMT and the final bounding box is determined, which maximizes the IoU term among annotators' manual bounding boxes.

Similarly, EPIC-KITCHENS-100 [127], expanded from EPIC-KITCHENS-55, contains 20M frames with more annotations of actions (90K), object bounding boxes, and hand bounding boxes. Furthermore, there are five additional participants for capturing videos. There are improvements in action annotations and narrations with more dense information. Even if EPIC-KITCHENS-55 solely contains object-bounding boxes, the new version additionally includes hand bounding boxes with left-and-right hand labeling. The annotations are inferred by automatic manners with a Mask R-CNN network [107] trained by the MS-COCO [132] dataset, and a hand-object interaction estimating network [52] trained on YouTube egocentric video frames.

5.1.3. HO-3D

HO-3D [36] is the first dataset with 3D annotations for both the hand and object in color images. It has already been updated to version 3 (HO-3D v3 [126]), but we mention only the original version (HO-3D v2 [36]). The difficulty in capturing hands without markers is that the hand and object annotations cannot be directly obtained from the observed scene. Thus, they proposed an automatic annotating method that utilizes MANO [49] to capture the hand. The dataset includes 10 objects from the YCB-video dataset [133]. There were 10 subjects who participated in capturing HO-3D.

From multiple numbers of RGB-D camera (Intel RealSense D415) sequences with certain frames, their 3D annotation method automatically labels a pose and shape of both a hand and an object at each frame. The hand model is represented by 45 MANO parameters and 6 transformation values, and the object mesh is brought from the YCB-video dataset. Then, the optimization method is used to annotate by minimizing the cost function which consists of data terms and constraint terms. The data terms include inconsistency terms (silhouette and depth), a 2D hand joint term, and 3D point term between hand and object point clouds. The constraint terms consist of anatomical constraints

Table 4
A summary of hand-object interaction datasets. Modality is how the data is generated in the real-world or synthetic world, and recorded in image or video. # images is a number of images or frames in datasets. # sub is a number of subjects participated in. Viewpoint is whether the data is captured in egocentric (first-view) or allocentric (third-view). Interaction info is what annotations are available in terms of hand-object interaction.

Dataset	Year	Type	Hand label[a]	Object label	Modality	# images	# sub/# obj	Scene	Camera intrinsic	Viewpoint	Interaction info[b]	Labeling method
Hands in action [115]	2012	RGB	Joint	Pose	Synth Video	8 sequences	–/3	–	O	allo	–	automatic
Yale Human Grasping [116]	2015	RGB	–	Class, Mass	Real Video	18K	4/11	Wild	O	ego	grasp [117], action	manual/automatic
GUN-71 [24]	2015	RGB-D	Joint, Seg	Seg	Real/Synth Image	12K	–/28	Wild	X	ego	contact, forces	manual/automatic
CMU grasp dataset [118]	2015	RGB	Seg	–	Real Image	945	1/2	In-lab	X	ego	–	automatic
Dexter+Object [29]	2016	RGB-D	Joint	Pose	Real Video	3014	–/1	In-lab	X	allo	–	manual
SynthHands [25]	2017	RGB-D	Joint	–	Synth Image	220K	–/7	–	O	ego	–	automatic
CORe50 [64]	2017	RGB-D	–	Class	Real Video	164K	–/50	Wild	X	ego	–	manual
EgoDexter [25]	2017	RGB-D	2D Joint, Joint	–	Real Image	3190	4/–	In-lab	X	ego	–	manual
Sth-Sth dataset [71]	2017	RGB	–	Class	Real Video	108K videos	1133/174	Wild	X	ego	action	manual/automatic
EPIC-KITCHENS-55 [65]	2018	RGB	–	B.B, Class	Real Video	11.5M	32/–	Wild	X	ego	action	automatic
VLOG [54]	2018	RGB	B.B	B.B	Real Video	114K	10.7k/30	Wild	X	allo	action	manual
YouCook2 [119]	2018	RGB	–	B.B	Real Video	2000 videos	–/–	Wild	X	ego, allo	action	manual
FPHA [32]	2018	RGB-D	Joint	Pose, Mesh	Real Image	105K	–/26	In-lab	X	ego	action	manual/automatic
Saudabayev et al. [120]	2018	RGB-D	–	–	Real Image	688K	13/	Wild	X	ego	grasp [105]	manual
ObMan [38]	2019	RGB-D	Joint, Mesh, Seg	Mesh, Seg	Synth Image	150k	–/2.7K	–	O	allo	–	automatic
ContactDB [121]	2019	Mesh	–	–	Mesh	–	−50/50	–	–	–	contact	automatic
HowTo100M [122]	2019	RGB	–	–	Real Video	136.6M	–	Wild	X	ego, allo	action	manual
Moments [123]	2019	RGB	–	–	Real Video	1M	–	Wild	X	ego, allo	action	automatic
HO-3D v2 [36]	2020	RGB-D	Joint, Mesh	Pose, Mesh	Real Video	80K	10/10	In-lab	X	allo	–	automatic
YCB-Affordance [104]	2020	RGB	Mesh	Mesh	Synth Image	133K	–/58	–	X	allo	grasp [105]	automatic
ContactPose [124]	2020	RGB-D	Joint	Mesh	Real Image	2.9M	50/25	In-lab	O	allo	contact, intent	automatic
100DOH [52]	2020	RGB	B.B	B.B	Real Image	100K	–	Wild	X	allo	contact	manual
GRAB [125]	2020	RGB	Mesh	Pose	Synth Video	953K	10/51	In-lab	X	allo	contact, intent	automatic
HO-3D v3 [126]	2021	RGB-D	Joint	Pose	Real Video	103K	10/10	In-lab	X	allo	–	automatic
DexYCB [76]	2021	RGB-D	Joint, Mesh	Pose, Mesh	Real Video	582K	10/20	In-lab	O	allo	–	manual/automatic
MOW [66]	2021	RGB	Joint, Mesh	Mesh	Real Image	512	450/121	Wild	X	ego, allo	contact	automatic
H2O [47]	2021	RGB-D	Joint, Mesh	Mesh	Real Image	572K	–/8	In-lab	O	ego	action	automatic
EPIC-KITCHENS-100 [127]	2021	RGB	B.B, Seg	B.B, Seg, Class	Real Video	20M	37/–	Wild	X	ego	action	automatic
H2O-3D [16]	2022	RGB-D	Joint, Mesh	Mesh	Real Image	76K	–/10	In-lab	X	allo	–	automatic
OakInk [128]	2022	RGB-D	Joint, Mesh	Mesh	Real Video	230K	12/100	In-lab	O	allo	action, contact	manual
AssemblyHands [129]	2023	RGB	Joint	–	Real Video	3.03M	34/–	In-lab	O	ego, allo	action	manual/automatic
ARCTIC [130]	2023	RGB	Joint, Mesh	Mesh	Real Video	2.1M	10/11	In-lab	O	ego, allo	contact, action	automatic

[a] Seg: segmentation map, B.B: bounding box on 2D image.

[b] grasp: grasp classes (taxonomies), action: a task with a certain object, contact: the contact region on the object and hand mesh.

about joint angles, the physical plausibility considered by the inter-penetration term regularizes the distance between a hand and object, and the temporal consistency between consequence frames.

5.1.4. *DexYCB*

DexYCB [76] is a markerless hand-object interaction dataset with 10 subjects and 20 kinds of objects from a YCB-video dataset [133]. There are 582K RGB-D frames captured by eight cameras (Intel RealSense D415) that record in different viewpoints. During the capturing stage of a scene, subjects encountered a table with 2 to 4 different kinds of objects, and they grasped the object five times. Especially, they were asked to pick up an object with the right hand in the first two trials, and then with the left hand in the next two trials. At the final trial, the randomized condition is used.

Their annotation process needs pre-defined keypoints of both hand and object. To get the keypoints, the researchers recruited annotators on Amazon Mechanical Turk (AMT) and the annotators manually labeled keypoints with VATIC [134] annotation tool. The annotators had to mark all 21 joint positions on the 2D image even if the joint position is not visible. In the case of object keypoints, two distinctive landmark points were marked: one is for a visible landmark and the other for an invisible landmark of the object at each frame.

For pose estimation, each hand model of the subject is pre-calculated with MANO, and each object mesh is brought from the corresponding mesh model in YCB-dataset. With pre-established keypoints and the meshes, the pose of the hand and object are regressed through optimization with a depth term, keypoints term, and regularization term like HO-3D [36] and Freihand [14]. The depth term quantifies the depth differences between the ground truth mesh and estimated point clouds of the hand and object. The keypoint term measures errors of estimated keypoints with respect to annotated keypoints in the projected space on a 2D image. Finally, the regularization term regularizes MANO pose parameters to stabilize the articulation of the hands.

5.2. *Synthetic datasets*

Real-world datasets are able to describe a human hand and the interaction between a hand and object in a realistic way, however, the ground truth annotations of the hand and object are not absolutely true due to their annotating methods. For instance, in the case of HO-3D [36], the annotating method is optimized with several optimizing terms, or conducted manually as in DexYCB [76]. In addition, human participants are needed to capture the *real* hands as well as a number of objects, so relatively small numbers of images are contained or there are limited types of hand grasping to interact with objects. In contrast, synthetic datasets can derive ground truth data directly from virtual scenes and do not have to recruit participants at all. We describe how the synthetic datasets are captured and how they generate a grasping hand pose with a simulation tool.

5.2.1. *ObMan*

ObMan [38] represents a synthetic *Object Man*ipulation dataset. The 2.7k object meshes are selected in the eight categories of ShapeNet dataset [135]: bottles, bowls, cans, jars, knives, cellphones, cameras, and remote controls. The hand mesh is generated by MANO [49]. To create a HOI synthetic dataset by GraspIt! [136], the hand needs to be a form of a rigid articulated model. Hence, the hand was comprised of 16 parts by dividing each finger into three parts and the palm as one part. For a realistic human hand image, the authors rendered not only the hand mesh but also a rendered arm connected to the hand using SMPL+H [49] model.

Physical-plausibility such as inter-penetration between a hand and object is an important aspect of hand-object interaction. For this, they utilized GraspIt!, a physical simulator that can ensure the physical-plausibility. When the simulated hand grasps the given object, GraspIt! assesses the stability of grasp based on two quality metrics, ϵ and υ [137]. The ϵ represents the minimum force and torque to contact points that make the object stable against external arbitrary forces. The υ quantifies the range of force and torque that the current grasp can endure. The feasibility of grasp is evaluated by comparing ϵ and υ. This metric is focused solely on the stability of the hand grasp, not measuring whether the grasp is natural in terms of hand-object interaction.

Based on a simulated hand-object mesh composite on the 3D space, 2D images were rendered with textures and backgrounds. The textures of objects are randomly brought from the texture map of ShapeNet, and the hand textures are acquired from full-body scan data in SURREAL [138]. However, the scans in SURREAL sometimes lack textures of the hand, thus a combined color was used with body textures and other subjects' hand colors. Finally, the background is filled with images that are sampled from LSUN [139] and ImageNet [140]. Notably, to guarantee that the hand-object appears in the image, they discarded images if the hand pixels were less than 100 pixels or less than 40% of the object was visible in the image.

5.2.2. YCB-Affordance

YCB-Affordance [104] is the extensive synthetic dataset with natural and realistic hand grasping in *multi*-object scenarios. It contains 133K frames, and the images are augmented on top of the YCB-video [133] dataset with synthetic grasping. The grasping hands were simulated by a GraspIt! [136] simulator, and there are 367 different grasping types within the grasp taxonomy [105].

To augment the YCB-video frame, a 6D object pose was first brought from the YCB-video dataset. Then, GraspIt! is able to generate a physically-plausible grasp to an object mesh. However, when the synthetic grasp is generated by the GraspIt! simulator, some grasps are not natural, despite they scored high in grasp metric [137]. For instance, this occurs when the simulated hand holds a knife blade or supports a cup with two fingers. In addition, while the grasp is available for one object, the hand may collide with other objects in the scene. Hence, the final dataset filters out the grasps of unnatural hand grasps and colliding with other objects.

5.3. Synthetic data generation module

We have reviewed real-world datasets and synthetic datasets of HOI scenarios. Real-world datasets often lack diversity in the HOI scenario, while synthetic datasets often lack validity in hand pose. Learning both the diversity and validity of hand pose from the vast training dataset is a challenging problem. To resolve this problem, Kailin Li et al. proposed ArtiBoost [114], an online synthetic data generation module that can be applied to an arbitrary hand-pose estimation framework. AritBoost generates synthetic images with varying hand poses and viewpoints for objects in a training dataset and resamples them during the training pipeline. In addition, to increase the variance in data distribution for generalization ability, they added noise to the hand pose and viewpoint direction. ArtiBoost attaches the texture to the variance-added hand and object mesh to generate rendered images. Those rendered synthetic images are sampled and mixed with the batch of a training dataset by a weight map which is updated each epoch. The weight map guides the sampling of hardly-discernable images more.

5.4. Dataset selection strategy

Several hand pose estimating approaches for hand-object interacting images have been trained using aforementioned datasets [32,36,38,65,76,104]. These datasets have been proven useful for hand-object pose estimation to train and evaluate the models. However, they may not provide sufficient training data to solve more complex problems, such as estimating contact information or classifying hand-object action. To address this point, we propose a brief strategy to select appropriate datasets for the hand-object pose estimation and other specific tasks.

When the target task is hand pose estimation from hand-object interaction images, HO-3D [36], DexYCB [76], and ObMan [38] are recommended. These datasets have been widely used in numerous methods, hence, the proposed method can be evaluated and compared with other approaches. However, they cannot cover all hand-object interacting situations, especially, when the human manipulates the articulated (or dynamic) objects. In this sense, ARCTIC [130] is recommended as it provides annotations for the articulated objects, such as a notebook and foldable cellphone. Although they can cover most situations, they lack natural hand-object interactions due to their controlled settings. To train the model with natural interactions, we recommend utilizing MOW [66] and EPIC-KITCHENS [65,127].

In addition to hand pose estimation, certain approaches have been developed to estimate hand-object interactions, including hand-object actions and contacts. Particularly, when a hand grasps an object, there exists a high correlation between the grasping and the object's features, such as its shape, function, or weight. For the hand grasp classifying task, the YCB-Affordance dataset [104] is advisable, annotated with the 33-grasping taxonomy [105]. Or, when the focus is on recognizing hand-object actions, we suggest employing OakInk [128], EPIC-KITCHENS [65, 127], and AssemblyHands [129]. Finally, when the objective is estimating the contact region between a hand and an object, contact-annotated datasets are necessary, for example, ContactDB [121], ContactPose [124], and ARCTIC [130].

6. Discussion

Our survey presents a comprehensive analysis of the existing approaches for estimating hand pose in hand-object interaction scenarios. Despite the extensive research in this field, we have identified several challenges that remain unresolved even with the emergence of deep learning-based methods. In this section, we explore these challenges and discuss potential future directions for improving hand pose estimation in human-object interaction (HOI) scenarios.

6.1. Challenges

6.1.1. Datasets

In Section 5, we provide an overview of datasets for hand-object interaction. Only some of real-world datasets [16,32,36,47, 66,76] are annotated for both the hand mesh and object mesh. Although object meshes can be easily annotated from existing data, such as YCB-video [133] and ShapeNet [135], hand meshes are hard to annotate through the markerless capturing process. These datasets [36,76] proposed an automatic labeling algorithm to reconstruct hand meshes with sparse labeling by person, given that the provided ground truth hand mesh is not perfect. In other words, the algorithm inevitably creates errors in predicting hand meshes. Also, real-world datasets often lack a variety of grasp types, even though they contain a lot of frames (images). Typically, they capture videos of interactions with objects where the grasp is maintained in the air. Moreover, they only contain interactions with a limited number of objects, around 10 [36] or 20 [76], which limits the variety of interactions.

Also, annotated real-world datasets have another limitation that the hand-object interaction is not natural. The acquisition stage of real-world datasets with comprehensive annotations necessarily requires a controlled capturing system, which limits the data to that collected in-lab, even though participants might be captured in a free environment. Therefore, to represent real hand-object interaction, a network must be trained on datasets collected in unconstrained environments such as EPIC-KITCHENS [65,127] and MOW [66].

While real-world datasets suffer from problems with labeling accuracy and a limited diversity of interactions, synthetic datasets [38,104] are not troubled by these issues. The annotation of hand and object can be directly acquired since synthetic datasets are collected from a simulated space. This provides more convenient and accurate information without miscalculations. Nonetheless, it leads to a domain gap in training networks due to differences between synthetic images and real-world images. Furthermore, while synthetic datasets offer the advantage of generating a large number of images with many objects (up to 2.7K [38]), they are limited in reflecting various aspects of hand-object interaction. These datasets primarily focus on stable grasping, as they are generated by physics simulation tools like GraspIt! [136] and rely on grasp metrics [137] commonly used in robotics for end-effectors. As a result, synthetic datasets may not accurately reflect real-world scenarios and the complexities of hand-object interaction.

6.1.2. Hand representations

Currently, most of pose estimating approaches rely on MANO [49] hand model. MANO provides a representation of the hand that has a high degree of freedom for expressive manipulation with its parameters. As it is derived from a data-driven manner with 1000 3D hand scans of 31 subjects, it can reflect prior knowledge of human hands well. However, MANO has shortcomings in that it cannot impose the physical and anatomical constraints of the hand.

To reflect the anatomical constraints of hand joints, Yang et al. [67] suggested A-MANO, anatomically constrained MANO hand model. A-MANO differs from MANO in terms of constrained angle movements. In their procedure, they first estimated a hand pose using MANO parameters and reconstructed the hand mesh from these parameters. Then, they adjusted the constraints on joint angles for certain keypoints of the MANO hand mesh. In addition, it is important to consider not only the angle of movement but the correlation of movement between fingers and phalanges. For instance, the ring finger cannot spread and move on its own when the middle finger is locked because it does not have independent tendons and muscles. Instead, its movement is facilitated by the activation of tendons connected to other fingers.

In contrast, other model-free hand representation, such as SDF, is advantageous to depict deformations or intrinsic kinematics of the hand. Hence, it is proper to utilize SDF when the target task is estimating a realistic hand shape. While MANO model can exploit the prior knowledge of the hand, however, SDF representation cannot utilize the prior knowledge. Nonetheless, AlignSDF [98] adopted both MANO and SDF, it cannot reflect the geometry, depicted in the input image [102].

6.1.3. Real-time requirement

There are applications that require hand pose estimation for tracking and recognizing gestures in virtual reality (VR). Those applications require short inference time for smooth operation. The high degrees of freedom from hand and object is the primary factor that leads to long inference time. As the complexity of a target increases, the network requires a bigger number of parameters. In addition, the refinement stage that ensures physical plausibility can also yield non-real-time performance in an iterative loop. Although Zhang et al. [90,95] presented efficient approaches using a depth image sequence of a hand and object, their implementation achieves only 25 fps which may not be sufficient for seamless application. Therefore, we are still in need for a lightweight network that can efficiently and accurately infer hand pose in real-time.

6.2. Future directions

6.2.1. Semi-supervised learning for unlabeled dataset

It is difficult to annotate the hand-object interaction dataset. The annotation of datasets needs hand and object labels including hand pose information, object classes, and action labeling for interaction. When the dataset targets dense pose estimation, the hand mesh and object mesh have to be labeled. To fit these two meshes manually involves exorbitant costs. Although real-world datasets such as HO-3D and DexYCB have proposed automatic labeling approaches, these cannot provide accurate annotation of hand and object meshes. To resolve labeling troubles, several semi-supervised learning approaches [33,42,60,69] utilize a part of the labeling data among otherwise unlabeled data or adapt the hand-only image domain to the hand-object interaction domain. These semi-supervised learning studies have proposed novel losses that can be calculated without supervision.

Furthermore, densely annotated datasets [36,76] offer abundant information, however, they are all captured in the laboratory setting. It differs from the wild datasets in terms of background, interacting objects, and grasping pose. The objective of hand-object pose estimation is to predict hand and object poses from the natural input of a camera. Therefore, networks trained on datasets that are limited to laboratory settings may struggle to accurately estimate hand-object poses in natural images.

6.2.2. Physically correct hand model

Existing hand pose estimation for hand-object interaction has employed the MANO [49] hand model, which is an excellent representation of dexterous manipulative hands with a high degree of freedom. However, these approaches do not properly reflect the physical properties of hands, as the MANO hand parameters were derived in a data-driven manner. Therefore, a physically correct hand model should be considered.

Before deep learning-based methods were introduced, Wu et al. [141] investigated the hand state by considering 32 possible hand configurations, based on whether each individual finger is bent or not. Then, they manually eliminated four hand states in which the pinky finger is bent while the ring finger is not, thus 28 anatomically possible hand state configurations are discovered. Despite their efforts, their hand state did not consider all possible hand configurations. Some generative methods [85,142,143] adopted a simple hand model from Oikonomidis et al. [142], which consisted of 37 basic geometric primitives such as cones, spheres, ellipsoids, and cylinders. Although this hand model was not similar to the real hand, such generative approaches could explicitly control the joint angles under a movable range of motion as part of the cost function during the optimization process. Nonetheless, they could not consider all physical properties of the hand, such as ring finger dependency.

These simple hand models can be a good starting point for developing a physics-based hand model, but they may not be suitable for deep learning-based hand pose estimation and current applications. In the future, we expect that researchers can explore the combinations of movement with realistic hand models. In other words, the network can regress the movement of the hand kinematic chain under physical conditions, before applying the MANO-based mesh skinning method along the skeleton. Another possible direction is using the existing method to regress the hand kinematic chain. Aristidou [144] proposed a hand model with physiological constraints such as ring finger dependency. He adopted an optimal marker-based approach and the final hand mesh is reconstructed by the inverse kinematic solver with physically-plausible movements for a natural and feasible hand.

6.2.3. Deformation of hand and object

The current hand-object interaction datasets do not consider the deformation of the hand or the object. The annotated hand mesh is not elastic, and most objects are rigid. Hence, the most of datasets depict the contact region as the intersection between the hand and the object. Although this representation could convey the contact information, it is not realistic or physically correct in terms of the deformation. The contact region is usually wider than the intersection region because of the deformation of hand tissue, and it is decided by deforming the hand and object, not penetrating each other. Furthermore, most approaches adopted the inter-penetration loss to regularize the volume of the intersection for ensuring physical-plausibility. However, if a network is trained on this kind of dataset, that contains the intersection between a hand and an object, the network may be confused about the meaning of the intersection: the training loss guides the network to a direction that decreases intersection during the training dataset contains the intersection.

Previously, Tsoli et al. [145] presented a tracking method that can follow a deformable object interacting with a hand in complex ways such as pushing, grasping, and folding cloth. However, their method can only deform a flat object such as clothing, and a hand is regarded as a rigid body. In addition, Pham et al. [146] and Hu et al. [147] suggested approaches that estimate the contact force between the hand and object based on an image. Although this cannot reflect an object deformation directly, the force estimation can be incorporated for object deformation.

Representing hand deformation is more challenging than representing object deformation due to the complex nature of the musculoskeletal structure. Moreover, while objects have their own template shapes and properties, human hands vary among individuals with differences based on sex, skeletal structure, and even occupation. Kadlevcek et al. [148] proposed a deformable template of the human body model that also varies among individuals as a human hand differs from. They simulate their template model using a physics-based method. Notably, changes in musculature can be represented by the preservation of muscle volumes, such as the contraction of the biceps muscle. Similarly, the deformation of hands can be calculated by volume preservation, incorporating the force estimation approaches and inter-penetration volume of the current hand-object mesh.

6.2.4. Feature fusion

It is a challenging task to extract information from a mutually occluded image, due to the lack of full vision of both the hand and the object. Previous approaches [74,80,93,97,149] have observed that the hand grasp itself has high correlation with the object's shape and features, as the palm side must touch the surface of the object. The occluded part also gives coherent information about nearby components. Additionally, if the object is too heavy to maintain a grasp on it, humans typically use a power grasp, in which the hand comes into contact with the palm. Therefore, studies have started to utilize hand features to estimate object poses and shapes, and vice versa. This is called feature fusion between a hand and object branch, and Chen et al. [93] have achieved better performance using it than with their baseline [38], non-fusion approaches.

Currently, feature fusion can be performed in the latent space, by concatenating the extracted features of the hand and object using an encoder before feeding them into the network body. Then, the network is back-propagated using a reconstructed hand pose (joint position or mesh) and reconstruction loss. Alternatively, one can perform feature fusion after generating the mesh using 3D estimating methods, such as AtlasNet [89] or GCN [40]. While feature fusion approaches for the hand and object achieved impressive performance, there is still room for improvement in terms of which specific features are fused.

To combine various hand and object features, an intermediate representation is necessary. In most feature fusion approaches, the extracted features estimate hand pose directly, making it difficult to determine what aspects the feature represents. For example, Wang et al. [74] used the extracted hand and object features to estimate hand joint positions and object silhouettes for supervision. This approach helps identify intermediate representations, resulting in more clearly separated representations in the extracted features. Even if features are used for both reconstructing the hand mesh and estimating joint positions, they are not separately extracted and fused.

To summarize, the feature fusion between hand and object branches may be a keystone to improve the performance of hand-object pose estimation network. Given a hand-object input image, several encoders may extract different aspects of the hand and object and all of these are potentially fused. For the case of the hand branch, a feature may represent grasp classification, joint position, or hand shape. In the context of the object, an extracted feature could involve object classification, shape, or usage.

7. Conclusion

In this paper, we provided a comprehensive overview of the current state-of-the-art techniques for estimating hand pose in hand-object interaction context. Our summary covers various deep learning-based approaches that estimate hand pose estimation in the forms of the hand joints and hand mesh including the object pose and shape. Additionally, we discussed the available datasets for hand-object interaction, including their annotations, features, and capture methods. Based on our survey, we identified several challenges and open research questions. We believe that this survey paper will provide valuable insights and knowledge for researchers interested in hand pose estimation from hand-object interaction images.

CRediT authorship contribution statement

Taeyun Woo: Conceptualization, Methodology, Validation, Investigation, Resources, Writing – original draft, Writing – review & editing. **Wonjung Park:** Conceptualization, Methodology, Resources, Writing – original draft, Writing – review & editing. **Woohyun Jeong:** Conceptualization, Methodology, Validation, Investigation, Writing – original draft. **Jinah Park:** Conceptualization, Methodology, Validation, Investigation, Resources, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work was supported by Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korea government (MOTIE) (20201510300280, Development of a REMOTE DISMANTLING TRAINING SYSTEM with force-torque responding virtual nuclear power plant).

Appendix

Evaluation metrics

For detailed explanations of evaluation metrics in the tables, we additionally list the metrics with explanations. The up arrow ↑ means that the metric should be increased for better performance. On the other hand, the down arrow ↓ implies the metric should be decreased for better performance.

Hand evaluation metrics

- Joint error (JE, ↓): it measures the summation of errors between 21 hand joint positions.
- Mean per joint position error (MPJPE, ↓): MPJPE also measures the average of errors between all 21 hand joints. Or, simply, it also called a mean joint error (MJE).
- Mean joint angle error (MJA, ↓): while the above two metrics calculate the positional error, MJAE calculates the angle differences at each joint.
- Mean mesh error (MME, ↓): it calculates the average difference between the ground truth mesh (hand or object) and estimated mesh.
- Percentage of correct keypoints (PCK, ↑): while other metrics show the average difference between estimation and ground truth, PCK shows the differences as percentages (%) in various threshold values. In other words, it shows how much the estimated joint positions (or vertices of a mesh) are well predicted under certain error thresholds.
- Area under curve (AUC, ↑): AUC, literally, represents an area under the curve. Usually, PCK graphs are used as the curve for AUC as PCK-AUC.

Object evaluation metrics

- Percentage of correct pose (PCP, ↑): to evaluate the correctness of the bounding box, PCP calculates the percentage of correct vertices of the bounding box based on error thresholds as same as PCK for a hand.
- Chamfer distance (CD, ↓): while the hand can be represented as the template model, such as MANO [49], there are various kinds of objects in the dataset. Hence, if the dataset contains many kinds of objects (*i.e.* ObMan [38] contains 2.7K object meshes), it is better to estimate the point cloud, instead of every mesh. In this case, Hasson et al. [38] estimate the point cloud of the object with AtlasNet [89], and the chamfer distance is used to quantify the differences between a ground truth point cloud and estimated one. To calculate the chamfer distance between two point clouds $S1$ and $S2$, the average distances of closest point pairs from $S1$ to $S2$, and from $S2$ to $S1$ are added.
- Average closest point distance (ADD-S, ↓) [133]: when the CD calculates the bidirectional distances of closest point pairs, ADD-S calculates the distances of closest point pairs between two point clouds in a unidirectional way.
- Percentage of average object 3D vertices error within 10% of object diameter (ADD-0.1D, ↑) [69]: it measures the percentage of vertices of estimated object mesh that are recorded less error than 0.1 diameters of the object.
- Maximum symmetry-aware surface distance (MSSD, ↓) [150]: MSSD calculates the difference between an estimated object mesh and the symmetric-transformed ground truth meshes. Unlike other object metrics, MSSD is less dependent on the object model and it can consider the symmetricity of objects. For more details, we recommend reading Hodaň et al. [150].

Hand-object interaction metrics

- Penetration depth (PD, ↓): PD can be a choice to measure the physical plausibility, especially in terms of penetration. It calculates how an estimated hand mesh is penetrated with respect to an object's surface as a length.
- Intersection volume (IV, ↓): another way to measure the penetration level, IV calculates the penetration as a volume of the intersected portion between a hand mesh and object mesh. For instance, Hasson et al. [38] measured IV by voxelizing the hand and object mesh with a voxel size as 0.5 cm.
- Classification accuracy (CA, ↑): in hand-object pose estimation, CA is used to measure how well the model can classify the hand-object interaction, such as grasping or pouring.
- Contact ratio (CR, ↑) [92]: when contact information is annotated, CR measures the percentage of how the contact is well covered based on annotated contact. Karunratanakul et al. [92] defined contact as any point on the hand's surface that is on or inside the object's surface between the hand and object. Then, they measured the ratio of contact that a point penetrates the surface more than zero.
- Disjointedness distance (DD, ↓) [67]: DD measures the stable contact between a hand and an object. DD is defined as the average distance of hand vertices in 5 fingertips to their closest object's surface [67].
- Simulation displacement (SD, ↓) [151]: SD is designed to evaluate the grasping quality from estimated hand pose. SD measures the average displacement of the object's center of mass in a simulated environment [152], assuming the hand is fixed and the object is affected by gravity. For more details, we recommend reading Tzionas et al. [151] and Hasson et al. [38].

References

[1] Lee T, Hollerer T. Multithreaded hybrid feature tracking for markerless augmented reality. IEEE Trans Visual Comput Graph 2009;15(3):355–68.

[2] Piumsomboon T, Clark A, Billinghurst M, Cockburn A. User-defined gestures for augmented reality. In: CHI'13 extended abstracts on human factors in computing systems. 2013, p. 955–60.

[3] Guleryuz OG, Kaeser-Chen C. Fast lifting for 3D hand pose estimation in AR/VR applications. In: 2018 25th IEEE international conference on image processing. IEEE; 2018, p. 106–10.

[4] Shi Y, Zhao L, Lu X, Hoang T, Wang M. Grasping 3D objects with virtual hand in VR environment. In: The 18th ACM SIGGRAPH international conference on virtual-reality continuum and its applications in industry. 2022, p. 1–8.

[5] Sharma A, Roo JS, Steimle J. Grasping microgestures: Eliciting single-hand microgestures for handheld objects. In: Proceedings of the 2019 CHI conference on human factors in computing systems. 2019, p. 1–13.

[6] Sharma A, Hedderich MA, Bhardwaj D, Fruchard B, McIntosh J, Nittala AS, et al. SoloFinger: Robust microgestures while grasping everyday objects. In: Proceedings of the 2021 CHI conference on human factors in computing systems. 2021, p. 1–15.

[7] Koppula HS, Saxena A. Anticipating human activities using object affordances for reactive robotic response. IEEE Trans Pattern Anal Mach Intell 2015;38(1):14–29.

[8] Antotsiou D, Garcia-Hernando G, Kim T-K. Task-oriented hand motion retargeting for dexterous manipulation imitation. In: Proceedings of the European conference on computer vision (ECCV) workshops. 2018.

[9] Sermanet P, Lynch C, Chebotar Y, Hsu J, Jang E, Schaal S, et al. Time-contrastive networks: Self-supervised learning from video. In: 2018 IEEE international conference on robotics and automation. IEEE; 2018, p. 1134–41.

[10] Li S, Ma X, Liang H, Görner M, Ruppel P, Fang B, et al. Vision-based teleoperation of shadow dexterous hand using end-to-end deep neural network. In: 2019 international conference on robotics and automation. IEEE; 2019, p. 416–22.

[11] Handa A, Van Wyk K, Yang W, Liang J, Chao Y-W, Wan Q, et al. Dexpilot: Vision-based teleoperation of dexterous robotic hand-arm system. In: 2020 IEEE international conference on robotics and automation. IEEE; 2020, p. 9164–70.

[12] Lopez PR, Oh J-H, Jeong JG, Jung H, Lee JH, Jaramillo IE, et al. Dexterous object manipulation with an anthropomorphic robot hand via natural hand pose transformer and deep reinforcement learning. Appl Sci 2023;13(1):379.

[13] Zimmermann C, Brox T. Learning to estimate 3D hand pose from single RGB images. In: Proceedings of the IEEE international conference on computer vision. 2017, p. 4903–11.

[14] Zimmermann C, Ceylan D, Yang J, Russell B, Argus M, Brox T. Freihand: A dataset for markerless capture of hand pose and shape from single RGB images. In: Proceedings of the IEEE/CVF international conference on computer vision. 2019, p. 813–22.

[15] Moon G, Lee KM. I2l-meshnet: Image-to-lixel prediction network for accurate 3D human pose and mesh estimation from a single RGB image. In: Computer Vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, Part VII 16. Springer; 2020, p. 752–68.

[16] Hampali S, Sarkar SD, Rad M, Lepetit V. Keypoint transformer: Solving joint identification in challenging hands and object interactions for accurate 3D pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 11090–100.

[17] Ahmad A, Migniot C, Dipanda A. Tracking hands in interaction with objects: A review. In: 2017 13th international conference on signal-image technology & internet-based systems. IEEE; 2017, p. 360–9.

[18] Oudah M, Al-Naji A, Chahl J. Hand gesture recognition based on computer vision: A review of techniques. J Imaging 2020;6(8):73.

[19] Huang L, Zhang B, Guo Z, Xiao Y, Cao Z, Yuan J. Survey on depth and RGB image-based 3D hand shape and pose estimation. Virtual Real Intell Hardw 2021;3(3):207–34.

[20] Rastgoo R, Kiani K, Escalera S. Sign language recognition: A deep survey. Expert Syst Appl 2021;164:113794.

[21] Goudie D, Galata A. 3D hand-object pose estimation from depth with convolutional neural networks. In: 2017 12th IEEE international conference on automatic face & gesture recognition. IEEE; 2017, p. 406–13.

[22] Choi C, Ho Yoon S, Chen C-N, Ramani K. Robust hand pose estimation during the interaction with an unknown object. In: Proceedings of the IEEE international conference on computer vision. 2017, p. 3123–32.

[23] Tompson J, Stein M, Lecun Y, Perlin K. Real-time continuous pose recovery of human hands using convolutional networks. ACM Trans Graph (ToG) 2014;33(5):1–10.

[24] Rogez G, Supancic JS, Ramanan D. Understanding everyday hands in action from RGB-D images. In: Proceedings of the IEEE international conference on computer vision. 2015, p. 3889–97.

[25] Mueller F, Mehta D, Sotnychenko O, Sridhar S, Casas D, Theobalt C. Real-time hand tracking under occlusion from an egocentric RGB-D sensor. In: Proceedings of the IEEE international conference on computer vision. 2017, p. 1154–63.

[26] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016, p. 770–8.

[27] Oberweger M, Wohlhart P, Lepetit V. Generalized feedback loop for joint hand-object pose estimation. IEEE Trans Pattern Anal Mach Intell 2019;42(8):1898–912.

[28] Jaderberg M, Simonyan K, Zisserman A, et al. Spatial transformer networks. Adv Neural Inf Process Syst 2015;28.

[29] Sridhar S, Mueller F, Zollhoefer M, Casas D, Oulasvirta A, Theobalt C. Real-time joint tracking of a hand manipulating an object from RGB-D input. In: Proceedings of European conference on computer vision. 2016.

[30] Tekin B, Bogo F, Pollefeys M. 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. 2019, p. 4511–20.

[31] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9(8):1735–80.

[32] Garcia-Hernando G, Yuan S, Baek S, Kim T-K. First-person hand action benchmark with RGB-D videos and 3D hand pose annotations. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 409–19.

[33] Baek S, Kim KI, Kim T-K. Weakly-supervised domain adaptation via GAN and mesh model for estimating 3D hand poses interacting objects. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 6121–31.

[34] Wei S-E, Ramakrishna V, Kanade T, Sheikh Y. Convolutional pose machines. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016, p. 4724–32.

[35] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial networks. Commun ACM 2020;63(11):139–44.

[36] Hampali S, Rad M, Oberweger M, Lepetit V. Honnotate: A method for 3D annotation of hand and object poses. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 3196–206.

[37] Zhang J, Jiao J, Chen M, Qu L, Xu X, Yang Q. A hand pose tracking benchmark from stereo matching. In: 2017 IEEE international conference on image processing. IEEE; 2017, p. 982–6.

[38] Hasson Y, Varol G, Tzionas D, Kalevatykh I, Black MJ, Laptev I, et al. Learning joint reconstruction of hands and manipulated objects. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 11807–16.

[39] Doosti B, Naha S, Mirbagheri M, Crandall DJ. Hope-net: A graph-based model for hand-object pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 6608–17.

[40] Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. 2016, arXiv preprint arXiv:1609.02907.

[41] Zhuang N, Mu Y. Joint hand-object pose estimation with differentiably-learned physical contact point analysis. In: Proceedings of the 2021 international conference on multimedia retrieval. 2021, p. 420–8.

[42] Cheng Z, Chen S, Zhang Y. Semi-supervised 3D hand-object pose estimation via pose dictionary learning. In: 2021 IEEE international conference on image processing. IEEE; 2021, p. 3632–6.

[43] Yin Y, McCarthy C, Rezazadegan D. Real-time 3D hand-object pose estimation for mobile devices. In: 2021 IEEE international conference on image processing. IEEE; 2021, p. 3288–92.

[44] Howard A, Sandler M, Chu G, Chen L-C, Chen B, Tan M, et al. Searching for mobilenetv3. In: Proceedings of the IEEE/CVF international conference on computer vision. 2019, p. 1314–24.

[45] Zhang M, Li A, Liu H, Wang M. Coarse-to-fine hand–object pose estimation with interaction-aware graph convolutional network. Sensors 2021;21(23):8092.

[46] Wen Y, Pan H, Yang L, Pan J, Komura T, Wang W. Hierarchical temporal transformer for 3D hand pose estimation and action recognition from egocentric RGB videos. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023, p. 21243–53.

[47] Kwon T, Tekin B, Stühmer J, Bogo F, Pollefeys M. H2O: Two hands manipulating objects for first person interaction recognition. In: Proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 10138–48.

[48] Sinha A, Choi C, Ramani K. Deephand: Robust hand pose estimation by completing a matrix imputed with deep features. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016, p. 4150–8.

[49] Romero J, Tzionas D, Black MJ. Embodied hands: Modeling and capturing hands and bodies together. ACM Trans Graph 2017;36(6):1–17.

[50] Gao H, Ji S. Graph U-Nets. In: International conference on machine learning. PMLR; 2019, p. 2083–92.

[51] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. Adv Neural Inf Process Syst 2017;30.

[52] Shan D, Geng J, Shu M, Fouhey DF. Understanding human hands in contact at internet scale. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 9869–78.

[53] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. Adv Neural Inf Process Syst 2015;28.

[54] Fouhey DF, Kuo W-c, Efros AA, Malik J. From lifestyle vlogs to everyday interactions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 4991–5000.

[55] Mittal A, Zisserman A, Torr PH. Hand detection using multiple proposals. In: Bmvc, vol. 2, no. 3. 2011, p. 5.

[56] Ohn-Bar E, Trivedi MM. Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations. IEEE Trans Intell Transp Syst 2014;15(6):2368–77.

[57] Bambach S, Lee S, Crandall DJ, Yu C. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In: Proceedings of the IEEE international conference on computer vision. 2015, p. 1949–57.

[58] Narasimhaswamy S, Wei Z, Wang Y, Zhang J, Hoai M. Contextual attention for hand detection in the wild. In: Proceedings of the IEEE/CVF international conference on computer vision. 2019, p. 9567–76.

[59] Huang L, Tan J, Meng J, Liu J, Yuan J. Hot-net: Non-autoregressive transformer for 3D hand-object pose estimation. In: Proceedings of the 28th ACM international conference on multimedia. 2020, p. 3136–45.

[60] Hasson Y, Tekin B, Bogo F, Laptev I, Pollefeys M, Schmid C. Leveraging photometric consistency over time for sparsely supervised hand-object reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 571–80.

[61] Hasson Y, Varol G, Schmid C, Laptev I. Towards unconstrained joint hand-object reconstruction from RGB videos. In: 2021 international conference on 3D vision. IEEE; 2021, p. 659–68.

[62] Kirillov A, Wu Y, He K, Girshick R. Pointrend: Image segmentation as rendering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 9799–808.

[63] Rong Y, Shiratori T, Joo H. Frankmocap: Fast monocular 3D hand and body motion capture by regression and integration. 2020, arXiv preprint arXiv:2008.08324.

[64] Lomonaco V, Maltoni D. CORe50: A new dataset and benchmark for continuous object recognition. In: Proceedings of the 1st annual conference on robot learning, vol. 78. 2017, p. 17–26.
[65] Damen D, Doughty H, Farinella GM, Fidler S, Furnari A, Kazakos E, et al. Scaling egocentric vision: The epic-kitchens dataset. In: Proceedings of the European conference on computer vision. 2018, p. 720–36.
[66] Cao Z, Radosavovic I, Kanazawa A, Malik J. Reconstructing hand-object interactions in the wild. In: Proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 12417–26.
[67] Yang L, Zhan X, Li K, Xu W, Li J, Lu C. CPF: Learning a contact potential field to model the hand-object interaction. In: Proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 11097–106.
[68] Qi CR, Su H, Mo K, Guibas LJ. Pointnet: Deep learning on point sets for 3D classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 652–60.
[69] Liu S, Jiang H, Xu J, Liu S, Wang X. Semi-supervised 3D hand-object poses estimation with interactions in time. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 14687–97.
[70] Lin T-Y, Dollar P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.
[71] Goyal R, Ebrahimi Kahou S, Michalski V, Materzynska J, Westphal S, Kim H, et al. The "something something" video database for learning and evaluating visual common sense. In: Proceedings of the IEEE international conference on computer vision. 2017, p. 5842–50.
[72] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. In: Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, Part III 18. Springer; 2015, p. 234–41.
[73] Moon G, Yu S-I, Wen H, Shiratori T, Lee KM. Interhand2. 6m: A dataset and baseline for 3D interacting hand pose estimation from a single RGB image. In: Computer vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, Part XX 16. Springer; 2020, p. 548–64.
[74] Wang R, Mao W, Li H. Interacting hand-object pose estimation via dense mutual attention. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. 2023, p. 5735–45.
[75] Moon G, Lee KM. I2L-MeshNet: Image-to-lixel prediction network for accurate 3D human pose and mesh estimation from a single RGB image. In: European conference on computer vision. 2020.
[76] Chao Y-W, Yang W, Xiang Y, Molchanov P, Handa A, Tremblay J, et al. DexYCB: A benchmark for capturing hand grasping of objects. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 9044–53.
[77] Fu Q, Liu X, Xu R, Niebles JC, Kitani KM. Deformer: Dynamic fusion transformer for robust hand pose estimation. 2023, arXiv preprint arXiv:2303.04991.
[78] Al-Rfou R, Choe D, Constant N, Guo M, Jones L. Character-level language modeling with deeper self-attention. In: Proceedings of the AAAI conference on artificial intelligence, vol. 33, no. 01. 2019, p. 3159–66.
[79] Bahat Y, Shakhnarovich G. Confidence from invariance to image transformations. 2018, arXiv preprint arXiv:1804.00657.
[80] Park J, Oh Y, Moon G, Choi H, Lee KM. Handoccnet: Occlusion-robust 3D hand mesh estimation network. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 1496–505.
[81] Kato H, Ushiku Y, Harada T. Neural 3D mesh renderer. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 3907–16.
[82] Taheri O, Ghorbani N, Black MJ, Tzionas D. 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. Springer; 2020, p. 581–600.
[83] Gu J, Bradbury J, Xiong C, Li VO, Socher R. Non-autoregressive neural machine translation. 2017, arXiv preprint arXiv:1711.02281.
[84] Oikonomidis I, Kyriazis N, Argyros AA. Efficient model-based 3D tracking of hand articulations using kinect. In: BmVC, vol. 1, no. 2. 2011, p. 3.
[85] Kyriazis N, Argyros A. Physically plausible 3D scene tracking: The single actor hypothesis. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2013, p. 9–16.
[86] Grady P, Tang C, Twigg CD, Vo M, Brahmbhatt S, Kemp CC. Contactopt: Optimizing contact to improve grasps. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 1471–81.
[87] Jiang H, Liu S, Wang J, Wang X. Hand-object contact consistency reasoning for human grasps generation. In: Proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 11107–16.
[88] Park JJ, Florence P, Straub J, Newcombe R, Lovegrove S. Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 165–74.
[89] Groueix T, Fisher M, Kim VG, Russell BC, Aubry M. A papier-mâché approach to learning 3D surface generation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 216–24.
[90] Zhang H, Bo Z-H, Yong J-H, Xu F. InteractionFusion: Real-time reconstruction of hand poses and deformable objects in hand-object interactions. ACM Trans Graph 2019;38(4):1–11.
[91] Bo Z-H, Zhang H, Yong J-H, Gao H, Xu F. DenseAttentionSeg: Segment hands from interacted objects using depth input. Appl Soft Comput 2020;92:106297.
[92] Karunratanakul K, Yang J, Zhang Y, Black MJ, Muandet K, Tang S. Grasping field: Learning implicit representations for human grasps. In: 2020 International conference on 3D vision. IEEE; 2020, p. 333–44.
[93] Chen Y, Tu Z, Kang D, Chen R, Bao L, Zhang Z, et al. Joint hand-object 3D reconstruction from a single image with cross-branch feature fusion. IEEE Trans Image Process 2021;30:4008–21.
[94] Almadani M, Elhayek A, Malik J, Stricker D. Graph-based hand-object meshes and poses reconstruction with multi-modal input. IEEE Access 2021;9:136438–47.
[95] Zhang H, Zhou Y, Tian Y, Yong J-H, Xu F. Single depth view based real-time reconstruction of hand-object interactions. ACM Trans Graph 2021;40(3):1–12.
[96] Zhou Y, Habermann M, Xu W, Habibie I, Theobalt C, Xu F. Monocular real-time hand shape and motion capture using multi-modal data. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 5346–55.
[97] Tse THE, Kim KI, Leonardis A, Chang HJ. Collaborative learning for hand and object reconstruction with attention-guided graph convolution. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 1664–74.
[98] Chen Z, Hasson Y, Schmid C, Laptev I. AlignSDF: Pose-aligned signed distance fields for hand-object reconstruction. In: Computer Vision–ECCV 2022: 17th European conference, Tel Aviv, Israel, October 23–27, 2022, proceedings, Part I. Springer; 2022, p. 231–48.
[99] Ye Y, Gupta A, Tulsiani S. What's in your hands? 3D reconstruction of generic objects in hands. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 3895–905.
[100] Aboukhadra AT, Malik J, Elhayek A, Robertini N, Stricker D. THOR-Net: End-to-end graformer-based realistic two hands and object reconstruction with self-supervision. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. 2023, p. 1001–10.
[101] Zhao W, Tian Y, Ye Q, Jiao J, Wang W. Graformer: Graph convolution transformer for 3D pose estimation. 2021, arXiv preprint arXiv:2109.08364.
[102] Chen Z, Chen S, Schmid C, Laptev I. gSDF: Geometry-driven signed distance functions for 3D hand-object reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023, p. 12890–900.
[103] Pavlakos G, Zhou X, Daniilidis K. Ordinal depth supervision for 3D human pose estimation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 7307–16.
[104] Corona E, Pumarola A, Alenya G, Moreno-Noguer F, Rogez G. Ganhand: Predicting human grasp affordances in multi-object scenes. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 5031–41.
[105] Feix T, Romero J, Schmiedmayer H-B, Dollar AM, Kragic D. The grasp taxonomy of human grasp types. IEEE Trans Hum-Mach Syst 2015;46(1):66–77.
[106] Song S, Xiao J. Deep sliding shapes for amodal 3D object detection in RGB-D images. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016, p. 808–16.
[107] He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. In: Proceedings of the IEEE international conference on computer vision. 2017, p. 2961–9.
[108] Garland M, Heckbert PS. Surface simplification using quadric error metrics. In: Proceedings of the 24th annual conference on computer graphics and interactive techniques. 1997, p. 209–16.
[109] Lorensen WE, Cline HE. Marching cubes: A high resolution 3D surface construction algorithm. ACM Siggraph Comput Graph 1987;21(4):163–9.
[110] Arnab A, Dehghani M, Heigold G, Sun C, Lučić M, Schmid C. Vivit: A video vision transformer. In: Proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 6836–46.
[111] Bertasius G, Wang H, Torresani L. Is space-time attention all you need for video understanding? In: ICML, vol. 2, no. 3. 2021, p. 4.
[112] Tkach A, Pauly M, Tagliasacchi A. Sphere-meshes for real-time hand modeling and tracking. ACM Trans Graph (ToG) 2016;35(6):1–11.
[113] Newcombe RA, Fox D, Seitz SM. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2015, p. 343–52.

[114] Li K, Yang L, Zhan X, Lv J, Xu W, Li J, et al. ArtiBoost: Boosting articulated 3D hand-object pose estimation via online exploration and synthesis. 2021, arXiv preprint arXiv:2109.05488.

[115] Ballan L, Taneja A, Gall J, Van Gool L, Pollefeys M. Motion capture of hands in action using discriminative salient points. In: Computer Vision–ECCV 2012: 12th European conference on computer vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12. Springer; 2012, p. 640–53.

[116] Bullock IM, Feix T, Dollar AM. The Yale human grasping dataset: Grasp, object, and task data in household and machine shop environments. Int J Robot Res 2015;34(3):251–5.

[117] Feix T, Pawlik R, Schmiedmayer H-B, Romero J, Kragic D. A comprehensive grasp taxonomy. In: Robotics, science and systems: workshop on understanding the human hand for advancing robotic manipulation, vol. 2, no. 2.3. Seattle, WA, USA;; 2009, p. 2–3.

[118] Saran A, Teney D, Kitani KM. Hand parsing for fine-grained recognition of human grasps in monocular images. In: 2015 IEEE/RSJ international conference on intelligent robots and systems. IEEE; 2015, p. 5052–8.

[119] Zhou L, Xu C, Corso J. Towards automatic learning of procedures from web instructional videos. In: Proceedings of the AAAI conference on artificial intelligence, vol. 32, no. 1. 2018.

[120] Saudabayev A, Rysbek Z, Khassenova R, Varol HA. Human grasping database for activities of daily living with depth, color and kinematic data streams. Sci Data 2018;5(1):1–13.

[121] Brahmbhatt S, Ham C, Kemp CC, Hays J. Contactdb: Analyzing and predicting grasp contact via thermal imaging. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 8709–19.

[122] Miech A, Zhukov D, Alayrac J-B, Tapaswi M, Laptev I, Sivic J. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In: Proceedings of the IEEE/CVF international conference on computer vision. 2019, p. 2630–40.

[123] Monfort M, Andonian A, Zhou B, Ramakrishnan K, Bargal SA, Yan T, et al. Moments in time dataset: One million videos for event understanding. IEEE Trans Pattern Anal Mach Intell 2019;42(2):502–8.

[124] Brahmbhatt S, Tang C, Twigg CD, Kemp CC, Hays J. ContactPose: A dataset of grasps with object contact and hand pose. In: The European conference on computer vision. 2020.

[125] Taheri O, Ghorbani N, Black MJ, Tzionas D. GRAB: A dataset of whole-body human grasping of objects. In: European conference on computer vision. 2020.

[126] Hampali S, Sarkar SD, Lepetit V. HO-3D_v3: Improving the accuracy of hand-object annotations of the HO-3D dataset. 2021, arXiv preprint arXiv:2107.00887.

[127] Damen D, Doughty H, Farinella GM, Furnari A, Ma J, Kazakos E, et al. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. Int J Comput Vis (IJCV) 2022;130:33–55.

[128] Yang L, Li K, Zhan X, Wu F, Xu A, Liu L, Lu C. OakInk: A large-scale knowledge repository for understanding hand-object interaction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 20953–62.

[129] Ohkawa T, He K, Sener F, Hodan T, Tran L, Keskin C. AssemblyHands: Towards egocentric activity understanding via 3D hand pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023, p. 12999–3008.

[130] Fan Z, Taheri O, Tzionas D, Kocabas M, Kaufmann M, Black MJ, et al. ARCTIC: A dataset for dexterous bimanual hand-object manipulation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023, p. 12943–54.

[131] Yuan S, Ye Q, Stenger B, Jain S, Kim T-K. Bighand2. 2 m benchmark: Hand pose dataset and state of the art analysis. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 4866–74.

[132] Lin T-Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, et al. Microsoft coco: Common objects in context. In: Computer vision–ECCV 2014: 13th European conference, Zurich, Switzerland, September 6-12, 2014, proceedings, Part V 13. Springer; 2014, p. 740–55.

[133] Xiang Y, Schmidt T, Narayanan V, Fox D. Posecnn: A convolutional neural network for 6D object pose estimation in cluttered scenes. 2017, arXiv preprint arXiv:1711.00199.

[134] Vondrick C, Patterson D, Ramanan D. Efficiently scaling up crowdsourced video annotation: A set of best practices for high quality, economical video labeling. Int J Comput Vis 2013;101:184–204.

[135] Chang AX, Funkhouser T, Guibas L, Hanrahan P, Huang Q, Li Z, et al. Shapenet: An information-rich 3D model repository. 2015, arXiv preprint arXiv:1512.03012.

[136] Miller AT, Allen PK. Graspit! a versatile simulator for robotic grasping. IEEE Robot Autom Mag 2004;11(4):110–22.

[137] Ferrari C, Canny JF. Planning optimal grasps. In: ICRA, vol. 3, no. 4. 1992, p. 6.

[138] Varol G, Romero J, Martin X, Mahmood N, Black MJ, Laptev I, et al. Learning from synthetic humans. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 109–17.

[139] Yu F, Seff A, Zhang Y, Song S, Funkhouser T, Xiao J. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. 2015, arXiv preprint arXiv:1506.03365.

[140] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. Imagenet large scale visual recognition challenge. Int J Comput Vis 2015;115:211–52.

[141] Wu Y, Lin J, Huang TS. Analyzing and capturing articulated hand motion in image sequences. IEEE Trans Pattern Anal Mach Intell 2005;27(12):1910–22.

[142] Oikonomidis I, Kyriazis N, Argyros AA. Full dof tracking of a hand interacting with an object by modeling occlusions and physical constraints. In: 2011 international conference on computer vision. IEEE; 2011, p. 2088–95.

[143] Kyriazis N, Argyros A. Scalable 3D tracking of multiple interacting objects. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2014, p. 3430–7.

[144] Aristidou A. Hand tracking with physiological constraints. Vis Comput 2018;34:213–28.

[145] Tsoli A, Argyros AA. Joint 3D tracking of a deformable object in interaction with a hand. In: Proceedings of the European conference on computer vision. 2018, p. 484–500.

[146] Pham T-H, Kyriazis N, Argyros AA, Kheddar A. Hand-object contact force estimation from markerless visual tracking. IEEE Trans Pattern Anal Mach Intell 2017;40(12):2883–96.

[147] Hu H, Yi X, Zhang H, Yong J-H, Xu F. Physical interaction: Reconstructing hand-object interactions with physics. In: SIGGRAPH Asia 2022 conference papers. 2022, p. 1–9.

[148] Kadleček P, Ichim A-E, Liu T, Křivánek J, Kavan L. Reconstructing personalized anatomical models for physics-based body animation. ACM Trans Graph 2016;35(6):1–13.

[149] Li D, Chen C. Tracking a hand in interaction with an object based on single depth images. Multimedia Tools Appl 2019;78:6745–62.

[150] Hodaň T, Sundermeyer M, Drost B, Labbé Y, Brachmann E, Michel F, et al. BOP challenge 2020 on 6D object localization. In: Computer vision–ECCV 2020 workshops: Glasgow, UK, August 23–28, 2020, proceedings, Part II 16. Springer; 2020, p. 577–94.

[151] Tzionas D, Ballan L, Srikantha A, Aponte P, Pollefeys M, Gall J. Capturing hands in action using discriminative salient points and physics simulation. Int J Comput Vis 2016;118:172–93.

[152] Coumans E, et al. Bullet real-time physics simulation. 2013, URL http://bulletphysics.org.