Robust Hand Pose Estimation during the Interaction with an Unknown Object

Chiho Choi Sang Ho Yoon Chin-Ning Chen Karthik Ramani
Purdue University
West Lafayette, IN 47907, USA
{chihochoi, yoon87, chen2300, ramani}@purdue.edu

Abstract

This paper proposes a robust solution for accurate 3D hand pose estimation in the presence of an external object interacting with hands. Our main insight is that the shape of an object causes a configuration of the hand in the form of a hand grasp. Along this line, we simultaneously train deep neural networks using paired depth images. The object-oriented network learns functional grasps from an object perspective, whereas the hand-oriented network explores the details of hand configurations from a hand perspective. The two networks share intermediate observations produced from different perspectives to create a more informed representation. Our system then collaboratively classifies the grasp types and orientation of the hand and further constrains a pose space using these estimates. Finally, we collectively refine the unknown pose parameters to reconstruct the final hand pose. To this end, we conduct extensive evaluations to validate the efficacy of the proposed collaborative learning approach by comparing it with self-generated baselines and the state-of-the-art method.

1. Introduction

Real-time depth data acquisition from commercial sensors has helped to simplify the tasks for hand pose estimation over the last decade. Although extensive research has been conducted on finding a robust and efficient solution for kinematic pose estimation of an isolated hand [14, 5, 6, 12, 29, 26, 17, 21, 2, 24, 22], the problem of the hand's interactions with a physical object is barely considered in the literature. The current approaches allow the user to manipulate a known object and a simple primitive shape such as a cylinder or cuboid. Therefore, these solutions do not work consistently with general human-computer interaction interfaces and augmented reality applications during natural interactions.

Hand pose estimation during the interaction with an unknown object is a challenging problem due to (i) the loss of hand information caused by partial or full object occlusions, (ii) the complicated shape of the unknown object and articulated nature of the hand, (iii) global 3D rotations, and (iv) the noise in acquired data, which confounds continuous estimation. In this paper, we present a new framework to effectively resolve these issues by collaboratively learning deep convolutional features from a hand and object perspective. Our fundamental observation from earlier work [20, 15] is that the interacting object can be a source of constraint on hand poses. In this view, we employ pose dependency on the shape of the object to learn discriminative features of the hand-object interaction.

The traditional approaches for pose estimation start with segmenting hand and object regions using RGB data followed by running an SVM classifier [18] or pixel-wise part classification [23] using hand-crafted features. A convolutional neural network (ConvNet) has recently been adopted to replace the hand-crafted features in [19], but this approach only aims for grasp classification. In contrast to these methods, we introduce a simultaneous training of deep neural networks for hand pose estimation. As a first step, we localize both the hand and object position using a ConvNet architecture. Specifically, we show that predicting the positions in the form of the heatmaps is an efficient way of overcoming the use of simple heuristics such as color-based segmentation or known object initialization.

We leverage the paradigm of *analysis by synthesis* and create a population of everyday human grasps. Similar to [19], the scope of hand-object interactions includes daily activities captured from an *egocentric* viewpoint. We adopt a 33-class taxonomy [3] to focus more on the shape of the hand grasp rather than the grasping motion [11]. The hand-object interactions are effectively mesh modeled with the corresponding hand pose parameters and grasp class labels. Although these synthetic depth images are easily simulated and accurately annotated, they do not explore artifacts (*e.g.*, noise and distortion) of real data captured from 3D sensors [22]. Thus, we design a fully unsupervised learning architecture to generate reconstructed data based on the idea of signal reconstruction in autoencoders. The output images are used to extract grasp features encoded in pairs -

one from a hand perspective and the other from an object perspective. To this end, we validate the use of two input sources (*i.e*. hand and object), in the context of grasp classification and consequently for hand pose estimation.

Our main contributions are summarized as follows:

1. Localization of an articulated hand and unknown object using a ConvNet architecture that directly regresses the heatmaps corresponding to the center position of the targets.
2. Use of object shape information as a latent cue to estimate a hand pose in the form of grasp classification.
3. Pixel-wise recreation of input data to correct the error of the sensor and mimic the attributes of synthetic data, which makes the system more robust.
4. A multi-channel pipeline to encode the grasp representations in pairs from an unknown object along with an observed hand.

In the remainder of this paper, Section 2 reviews relevant literature on 3D hand pose estimation and hand-object interaction. In Section 3, we present a localization network to segment object and hand regions. Section 4 describes the creation of a synthetic dataset and the reproduction of a realistic dataset. Subsequently, we discuss our novel architecture for hand pose estimation during the interaction with an unknown object in Section 5. Section 6 validates our system from evaluations. In the last section, conclusions are presented.

2. Related Work

In this section, we review some work relevant to depth camera- or RGB-D input-based approaches.

2.1. Pose estimation of an isolated hand

Generative approaches (model-based) The optimization of an objective function is proposed to recover the hand configuration using a 3D hand model. Initially, particle swarm optimization (PSO) [14] and a Gauss-Seidel solver [12] are used to guide the optimization toward the best solution. Although these methods for finding an alignment between models are straightforward, they require precise initialization at the beginning and may fail to track the hand when a prior estimate is inaccurate. Recently, the paradigm has shifted to reinitializing the population of the hand poses using external sources to mitigate the effect of model drift [32, 21, 7].

Discriminative approaches (appearance-based) A mapping between image features and corresponding pose configurations is learned for hand pose estimation [5, 6, 27]. However, these methods are susceptible to self-occlusions and self-similarities of the fingers. To alleviate a large error in the presence of occlusions, some approaches locally regress the pose parameters [24, 26, 28]. A collaborative filtering framework is proposed in [2] to regress the joint angle parameters from a set of similar poses. Subsequently, cascaded convolutional neural networks are trained to output deep activation features in [22] to improve the robustness upon occlusions and jitters by replacing hand-crafted features. In [4, 29], the 2D heatmaps corresponding to joint positions are regressed and 3D hand poses are recovered using a single-view or multi-view ConvNet.

2.2. Pose estimation during hand-object interaction

Previous approaches for hand pose estimation in hand-object interaction have mainly focused on model-based pose optimization [1, 8, 9, 15, 31], similar to generative methods in hand tracking. Some of these approaches aim to track the interacting hands from a multi-camera input with a manual initialization of a hand and object [1, 15, 31]. Even though a dynamics simulator [8] and an ensemble of collaborative trackers [9] are presented to handle multiple object tracking from a single RGB-D sensor, all these methods assume that the accurate 3D models of the manipulated objects are given. In [16], tracking hands in interaction with unknown objects is proposed for model reconstruction. However, their use of temporal information from a model-based hand tracker may cause a model drift and limit the functional range of hand-object interaction. Although our method also focuses on interaction with unknown objects, we do not explicitly track the object but try to learn a discriminative cue for hand pose estimation.

Besides these studies, our work shares similarities with [20, 15] in terms of pose dependency on the shape of the object. However, the method in [15] does not explicitly extract shape information from the object. In [20], a set of synthetic hand templates is used to find a similar pose while searching the nearest neighbor. However, the small number of examples in the database and the search complexity of this method are the major bottlenecks. Even though our method shares a similar insight, the search complexity is remedied by reducing the search space based on the grasp type and the orientation of the hand. Recently, [18, 23] have used hand-crafted features for pose estimation while interacting with an object. They first segment the hand and object regions using RGB data, and then run either an SVM classifier [18] or pixel-wise part classification [23] for hand pose estimation. However, these methods oversimplify the pose estimation problem by transferring a grasp template [18] or require a simple primitive as a manipulating object [23]. Even though a convolutional neural network framework is subsequently employed to replace the hand-crafted features [19], this approach only aims for grasp classification. In contrast, our method introduces a new ConvNet architecture effectively designed to handle the hand-object interaction for pose estimation that learns discriminative grasp features

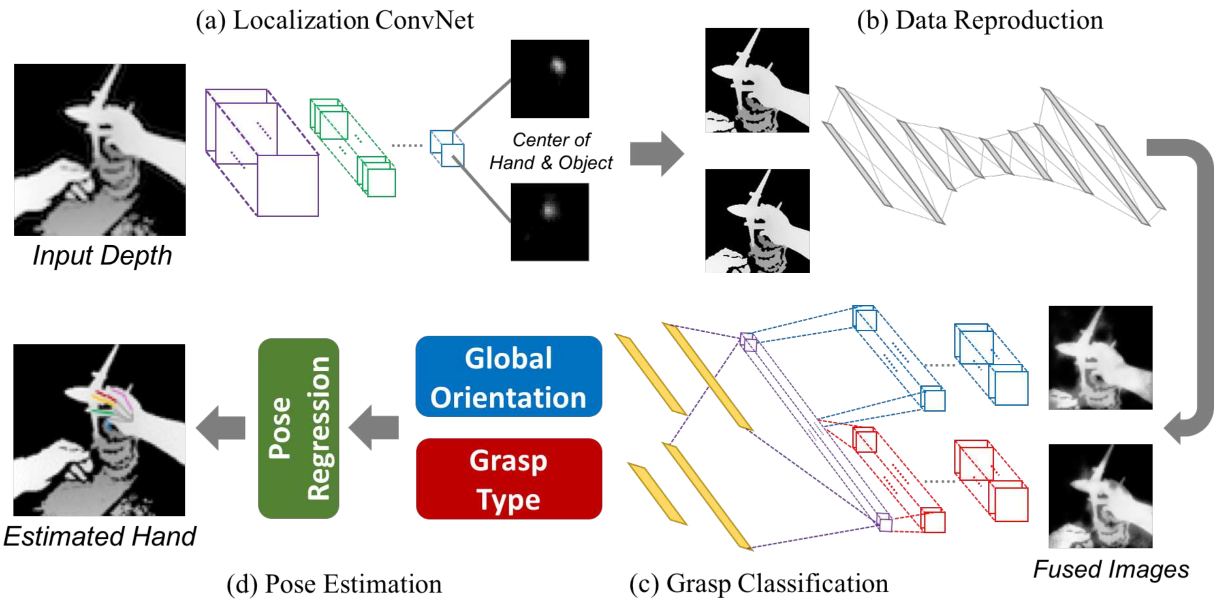


Figure 1: An overview of the proposed approach. (a) The localization ConvNet takes a depth image as input to predict the heatmaps of the hand and object center. (b) The reproduction network generates the informative fused images for grasp classification. (c) Our system collaboratively classifies both the global orientations of the hand and grasp type using the paired images. (d) Then, pose regression is applied to estimate the pose parameters of the hand.

from both perspectives (*i.e.*, of both the hand and the object). The pipeline overview is presented in Figure 1.

3. Hand-Object Localization

In this section, we first discuss a creation of our synthetic dataset that simulates the hand interacting with an object. Then we present our pragmatic solution to extract a center position of both the hand and the object for later use.

3.1. Synthetic dataset

3D hand Our hand model has a structure similar to that of the 21 DOFs kinematic mesh broadly used for hand pose estimation [2, 22]. We additionally construct the 2 DOFs lower arm to independently model the arm segment rotations, which helps to identify the global hand orientation, thus regularizing the jitter of the estimated pose [21]. Our training dataset simulates hand-object interaction from an entire egocentric viewpoint by rotating 3 wrist angles $\boldsymbol{\theta^W} = \{\theta_r^W, \theta_p^W, \theta_y^W\}$ where $\theta_r^W \in [-60, 60]°, \theta_p^W \in [-90, 90]°, \theta_y^W \in [-10, 50]°$. These rotational ranges are further quantized into the 48 orientation classes ($4{\times}6{\times}2$).
3D CAD models We collect 3D mesh models of 600 daily objects that can be easily obtained online[1] and are freely downloadable. Our object models are all rigid shapes and we only explicitly determine the contact points of each object for the specific grasp.

Dataset creation Manual simulation of hand-object interaction from different individuals is an unsupervised and time-consuming task that cannot even guarantee the annotation quality of the grasps. Along this line, we employ a model fitting method to optimize hand grasps with respect to the shape of the target objects. For this, particle swarm optimization is used to minimize the distance error between the observed object and our 3D hand model. Although this generative method guides the objective function to best fit the observed data, it might be susceptible to a collision of two geometric shapes (*i.e.*, the intersection of two triangular meshes). Therefore, we adopt a technique of collision detection to quickly determine if the grasp state is invalid. Details of collision detection are skipped for brevity, and we refer the readers to [10]. In practice, our approach reaches realistic object grasps and outputs the corresponding joint angle parameters of the hand with the grasp class label. We then insert these rendered depth maps into the cluttered background captured in-the-wild using Intel's RealSense F200, very similarly to [18]. This process is used not only to mimic an everyday environment for our simulated interaction but also to generalize our deep neural network - in particular, to handle the sensitiveness to diverse background perturbations. In total, we generate 330K synthetic depth maps. They are rendered from 33 grasps in

[1] 3D ContentCentral (https://www.3dcontentcentral.com) and GrabCAD (https://grabcad.com)

	Layers	# Kernels	Filter size	Stride	Pad
1	Conv	16	5×5×1	1	2
2	ReLU				
3	Pmax			2	0
4	Conv	32	5×5×16	1	2
5	ReLU				
6	Pmax			2	0
7	Conv	64	5×5×32	1	2
8	ReLU				
9	Pmax			2	0
10	Conv	128	5×5×64	1	2
11	ReLU				
12	Conv	256	5×5×128	1	2
13	ReLU				
14	Conv	2	5×5×256	1	2
15	ReLU				
16	L2				

Table 1: The design of a ConvNet for heatmap regression. (Conv: convolutional layer, Pmax: max pooling layer, ReLU: rectified linear units layer, L2: Euclidean loss layer)

terms of 40 objects (on avg.), 48 wrist rotations, and 5 populations per grasp[2].

3.2. Localization network

A heuristic method [17, 25, 22] to extract the region of interest cannot work consistently with general human-computer interaction applications. Hence, we train a ConvNet model to regress the confidence map (*i.e.*, the heatmap) of the center for the hand and object model (see Figure 2). Our fully convolutional network is comprised of six convolutional layers followed by a nonlinear layer. Furthermore, a final Euclidean loss layer computes the sum of squares of differences between the predicted heatmap and ground truth, as shown in Table 1. Even though the use of additional layers slightly increases estimation accuracy, the performance improvement is trivial compared to a significant increase in computation requirement.

Table 2 shows the quantitative comparison with a random forest (RF) classifier used in [29] which performs pixel-wise hand segmentation. Here, we first compute a centroid of segmented hand pixels and calculate the error in *pixels* from a centroid of ground truth. In contrast, our heatmap regressor directly outputs the position of the hand center and significantly outperforms the RF-based approach from localization accuracy.

Data Processing The depth values of input depth map D_m are first normalized to the range of [0, 255] to generate depth image D_i, and then we rescale D_i to width of 240. The rescaled depth image D_r of size 240×240 is fed into localization ConvNet. The network outputs two 30×30 heatmaps corresponding to the centroid of the hand and the object, respectively. Next, we up-sample these heatmaps with a scaling factor 8 and then rescale to width of 320 so that the size to be the same as the original depth map D_m. The maximum value in each heatmap marks the hand centroid $\{u_h, v_h\}$ and the object centroid $\{u_o, v_o\}$. Note that the depth value of these points $d_m^h = \{u_h, v_h, d_h\}$ and $d_m^o = \{u_o, v_o, d_o\}$ can be obtained from the original depth map D_m. We use d_m^h and d_m^o to generate 64×64 depth images D_i^h and D_i^o centered at the hand/object centroid. The above process is detailed in the supplementary material.

Model	Error	Settings
Ours	6.7 *pixels*	9 Epochs
RF [29]	27.6 *pixels*	22 Depth, 70 Trees

Table 2: Accuracy comparison of hand localization on our synthetic dataset.

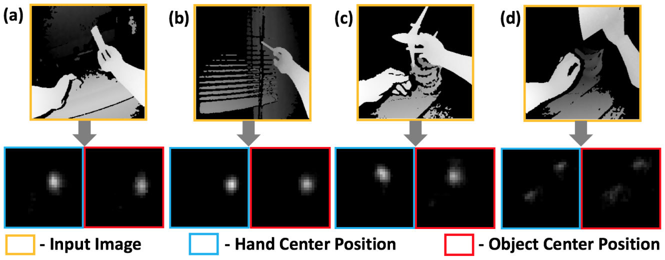


Figure 2: The heatmap regressor successfully segments the center points within contact regions for the hand and the object respectively (a)∼(c). The performance is lower in special cases such as introducing another hand in the scene (d).

4. Reproduction of Realistic Dataset

One observation obtained from quantitative evaluations from earlier work [22] is that the system of *analysis by synthesis* showed different aspects depending on the type of dataset. They evaluated their approach using synthetic and realistic datasets for self-comparison and comparison with the state-of-the-art, respectively. However, the system showed much better performance using a synthetic dataset. Even though [22] tried to mimic the actual sensor image by adding a Gaussian noise, there exists a gap between the two to be further improved. To address it, we propose a framework that allows the datasets to learn the attributes across domains instead of heuristically adding artifacts to the datasets or removing artifacts from them.

4.1. Synthesizing data by reconstruction

Our system is trained on a synthetic dataset that is virtually simulated with 3D mesh models. Although this ap-

[2] 33 grasps × 40 objects × 48 rotations × 5 populations ≈ 330K

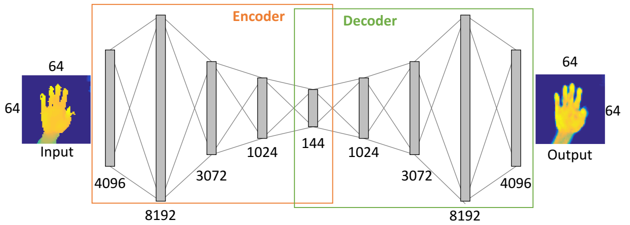


Figure 3: Overall architecture of the proposed data reproduction network.

proach is attractive because it allows the system to be applied to a range of sensor types, we might lose a certain degree of accuracy compared to the case when the same dataset type is used for both training and testing. Therefore, we generate synthesized real data based on the idea of signal reconstruction in autoencoders. The autoencoders try to predict the missing part from the non-missing values to recover original data. Our insight is that the loss of real data can be better represented by imposing the *repairing* process of an autoencoder. For this, we train our model to reconstruct pixel-level artifacts of the input depth, D_i^h and D_i^o.

In hand tracking literature, a synthesizer is proposed to correct the error of initial estimation in [13]. The initial pose estimation is used to generate a synthesized hand depth image, and the updater predicts an updated hand pose using both input data and the synthesized model in a closed loop. For this, they trained three different ConvNet models using a set of annotated training pairs. In contrast to this work, our approach differs as follows: (i) our method is unsupervised and we do not require any training pairs between real and synthetic data; (ii) we re-generate the synthesized depth image in a single shot without using inefficient iterations; and (iii) pixel-level noise and artifacts are tractable by encoding the input data and mapping back to the original data.

4.2. Reproduction network

Our system follows the traditional autoencoder framework which consists of two components, an *encoder* and a *decoder*. The encoder tries to reduce the dimensionality of the input by mapping high-dimensional data into a lower dimensional feature space, whereas the decoder recovers the original input by mapping back the learned representation into a high-dimensional space. The overall specification of our data reproduction network is displayed in Figure 3. We impose four hidden layers followed by a nonlinear function (sigmoid layer) for both the encoder and the decoder. The proposed network is trained on the 240K depth images captured across sensor types[3] and converged after 20 epochs.

In Figure 4, three data types are visually compared. The top row shows the *original* 64×64 depth images (D_i^h and D_i^o) selected from an NYU dataset [29] for proof of concept. The second row shows the corresponding *synthesized* images generated using our reproduction network. We note that pixel-wise artifacts (*e.g.*, holes or missing pixels) of the *original* images are eliminated from the *synthesized* images by the reconstruction process of the network. However, a new compression distortion is observed from the palm regions of the *synthesized* images. To further eliminate such distortions, we spatially *fuse* the depth images by averaging the input (*original*) and output (*synthesized*) images. This is a simple yet effective strategy to improve the overall performance. The improvement of classification accuracy (37.75% to 41.00% in Table 3) on the *fused* images (D_f^h and D_f^o) demonstrates the impact of the averaging process. We discuss more details with empirical validation in Section 6.

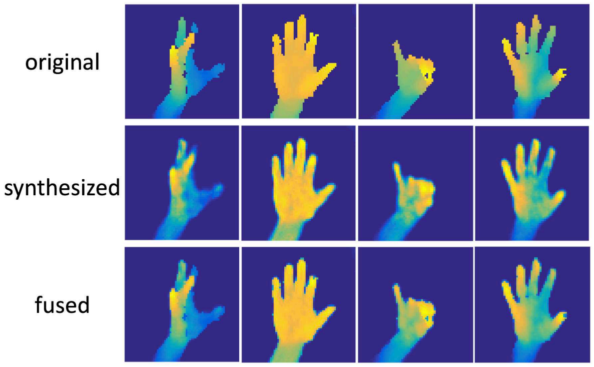


Figure 4: Visual comparison for data synthesis on the selected depth images of NYU dataset [29]. First row: the original depth images. Second row: the synthesized images using our framework. Third row: spatially fused images.

5. Hand Pose Estimation

5.1. Grasp classification

The partial or full loss of hand information during the interaction with hands cannot be recovered particularly when unknown objects are introduced. Instead of processing low-level data to recover or remove the region of object occlusions, we draw a ConvNet framework to extract informative expressions of grasps from those regions. We assume that there is a strong relation between the shape of the object and the configuration of the hand poses in the context of hand grasp. Thus, our model collaboratively learns the convolutional features about grasps from a hand and object perspective in pairs by sharing intermediate representations between two networks in the feature space.

Details of our network structure are shown in Figure 5. The *fused* 64×64 image pair (D_f^h and D_f^o) from the previous step is now used as input to this model. Each network independently learns discriminative representations from

[3] 80K synthetically rendered images + 160K real depth images (80K captured from PrimeSense & 80K from Intel's RealSense F200.)

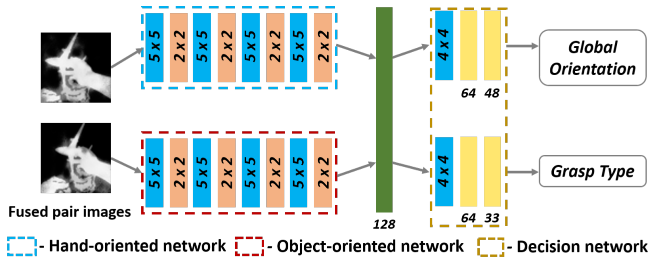


Figure 5: The architecture of proposed grasp classification network. Given fused pair images, each image is passed through distinctive networks to classify both hand's global orientation as well as grasp type. Color codes: Blue = Conv+ReLu, orange = Pmax, green = concatenation, yellow = Fully connected layer (ReLU exists between fully connected layer).

different perspectives: the hand-oriented network focuses on the loss of hand information caused by occlusions due to the object, while the object-oriented network extracts potential pose information even from the unseen object. Each feature map of size $4 \times 4 \times 64$ independently extracted after the fourth convolutional layer is then concatenated as a tensor of size $4 \times 4 \times 128$. This step is important to transfer knowledge about a perceptual set of attributes such as hand/object occlusions, shape, or silhouette learned from different domains. This vector is further used to estimate the pose parameters in the next subsection.

5.2. Pose estimation

Although the ConvNet-based hierarchical classification strategy is effective for finding unknown pose parameters [22], it is computationally inefficient to train every five networks corresponding to each of the 144 global bins. Our pose estimation method is inspired by a 2-stage hierarchical strategy, but we do not estimate the global parameters from stage 1. Instead, we only constrain the pose configuration space using the possible hand orientations and a grasp type likely to be a set of good initializations. Once we identify the reduced subset, then we evaluate all the pose parameters in an all-in-one approach in stage 2 from this space.

The decision network (5th convolutional layer and the following fully connected layers in Figure 5) first classifies the top 5 orientations using the softmax function. Our rationale for classifying the orientation of the hand is as follows: the overall performance of hand pose estimation becomes deterministic based on the robustness of pose initialization [21, 24], and the majority of the pose error is associated with the global orientation of the hand in practice. We subsequently classify the top 1 grasp type from the same network. Then we identify a reduced subset (*i.e.*, 1 grasp×40 objects×5 orientations×5 populations $\approx$ 1K) from our 330K training images. An additional 64-dimension feature vector $\mathbf{f_2}$ is extracted in the penultimate layer of the orientation decision network, which contains discriminative cues sufficient to classify the global orientation of the hand. Finally, we perform a nearest neighbor search from the restricted space to retrieve l poses similar to the input hand pose. In practice, we observe that the use of more neighbors does not effectively increase the overall performance but introduces a computational bottleneck.

Our regression method aligns with the collaborative learning approach [2, 22] to predict the pose parameters. Let $n = 64$ be a dimensionality of the feature vector, $m = 18$ be a number of joint angles, and $l = 32$ be a number of nearest neighbors, then the matrices $\mathbf{F_1} \in \mathbb{R}^{l \times n}$, $\mathbf{f_2} \in \mathbb{R}^{1 \times n}$, $\mathbf{P_1} \in \mathbb{R}^{l \times m}$, and $\mathbf{p_2} \in \mathbb{R}^{1 \times m}$ are the submatrices of M:

$$\mathbf{M} = \begin{bmatrix} \mathbf{F_1} & \mathbf{P_1} \\ \mathbf{f_2} & \mathbf{p_2} \end{bmatrix}, \tag{1}$$

where $\mathbf{F_1}$ is the feature vectors of neighboring poses, $\mathbf{f_2}$ is the feature vector of the current pose, $\mathbf{P_1}$ is the joint angles of neighboring poses, and $\mathbf{p_2}$ is the unknown angles to be regressed. We compute $\mathbf{p_2}$ using MacDuffees theorem:

$$\mathbf{p_2} = \mathbf{f_2}(\mathbf{F_1})^{+}\mathbf{P_1}, \tag{2}$$

where + denotes the Moore-Penrose pseudo-inverse. The proof of the above process is detailed in [22].

6. Experiments

We conduct extensive evaluations to verify our design choices for localization and grasp classification as well as hand pose estimation. To demonstrate the efficacy of our approach, we compare the results of testing our method and a state-of-the-art method using a public dataset and of testing our self-generated baselines using a synthetic dataset.

6.1. Datasets for comparison

The size of our synthetic dataset is 16.5K; it is comprised of 500 depth maps per grasp randomly rendered from different objects, orientations, and backgrounds. This dataset is used for comparison with self-generated baselines (described below) to validate our design choices. Since we aim to achieve 3D hand pose estimation, our dataset is fully annotated with the grasp numbers, orientation labels, joint angle parameters, and joint positions in 3D.

For localization and grasp classification, we additionally evaluate using a publicly available GUN-71[4] dataset [19]. It was captured in-the-wild from eight subjects covering 28 everyday objects per grasp with various *egocentric* views. Since the grasp type is labeled on a per-frame basis, it is

[4] Although GUN-71 dataset contains 71 grasps, we only use the common 33 grasps ($\approx$ 6K depth maps).

Test set / Train set	Original GUN-71	Synthesized GUN-71	Fused GUN-71
Original	39.75%	16.87%	31.71%
Synthesized	32.86%	37.75%	36.51%
Fused	36.43%	29.31%	**41.00%**

Table 3: Grasp classification results for 33 grasps evaluated on GUN-71 dataset [19]. The use of reproduction network (spatially fused) improves overall classification results. Note that *Train* denotes the type of training dataset used to train our model and *Test* denotes the format of GUN-71 dataset used for testing our networks.

suitable to evaluate the performance of the proposed reproduction network and grasp classification approach with respect to grasp recognition accuracy.

Although we are aware of the publicly available hand-object datasets in the literature [30, 23], we do not use them for evaluations. As we discussed in Section 3.1, our hand-object interactions are simulated from an *egocentric* viewpoint which differs from their interaction ranges. In addition, their software is not publicly available so we evaluate the quantitative/qualitative performance of our method using the synthetic dataset and the publicly available GUN-71.

6.2. Analysis of design choices

Experiments on public dataset To demonstrate the efficacy of the proposed data reproduction process, we individually train nine models using different types of dataset from scratch. We first define three types of training and test datasets: (i) *Original* denotes the original depth images (D_i^h and D_i^o) obtained as a result of localization; (ii) *Synthesized* is a set of images outputted from the reproduction network; (iii) *Fused* indicates the images (D_f^h and D_f^o) obtained by spatially averaging the *Original* and *Synthesized* data. The experimental result is shown in Table 3. The best performance (accuracy of 41.00 %) is achieved when the network is trained using the spatially fused images and tested on the same type of dataset. It validates that training and testing with *Fused* data allows the extraction of more expressive representations of data while minimizing depth artifacts. Interestingly, the model that is trained and tested using the *Synthesized* data shows poorer performance than the model that is trained and tested using the *Original* data. Here we observe that the higher accuracy may not be accomplished by simply synthesizing the depth images because the reproduced dataset could explore a new distortion, as also shown in Figure 4 (second row). Subsequently, Table 4 compares the performance of our grasp classification method to that of [19]. Note that the accuracy of [19] is directly captured from their paper. All our methods significantly outperform their deep feature-based SVM grasp classifier by a huge margin. This comparison validates the rationale of our specific approach against other choices.

Model	Classification accuracy
Rogez et al. [19]	20.50 %
Original	39.75 %
Synthesized	37.75 %
Ours (*Fused*)	**41.00 %**

Table 4: Accuracy comparison of grasp classification on GUN-71 dataset.

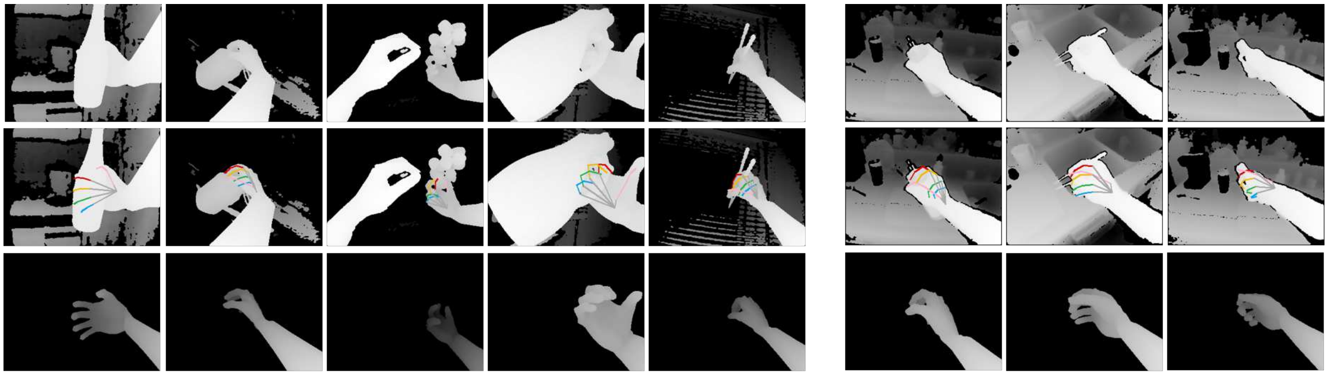
Network	*Hand-only*	*Ojbect-only*	**Ours**
Orientation Acc.	59.31%	51.12%	**60.50%**
Grasp Acc.	43.87%	49.12%	**55.56%**

Table 5: Classification accuracy for the orientation of the hand and the grasp type. *Hand only* achieves higher performance to orientation classification than *Object only* but has less impact on grasp classification.

Experiments on synthetic dataset We conduct more ablative tests that demonstrate the efficacy of our two-stream (the hand and object stream shown in Figure 5) orientation/grasp classification network. For this, we compare our two-stream network to two additional baselines by conducting tests with a synthetic dataset: (i) with only the hand stream (*Hand-only*) and (ii) with only the object stream (*Object-only*). Table 5 shows the performance of these baselines relative to our proposed approach. As expected, the *Hand-only* stream performs better to classify the orientation of the hand, whereas the *Object-only* stream achieves higher accuracy for grasp type classification relative to the *Hand-only* stream. It implies that the *Hand-only* stream extracts more beneficial information about the configuration of the hand. The *Object-only* stream focuses more on the shape of the object, which infers hand grasp. The proposed two-stream strategy outperforms these two baselines by extracting informative representations from both streams. It validates that constructing the two-stream network is critical to good performance.

6.3. Evaluation for pose estimation

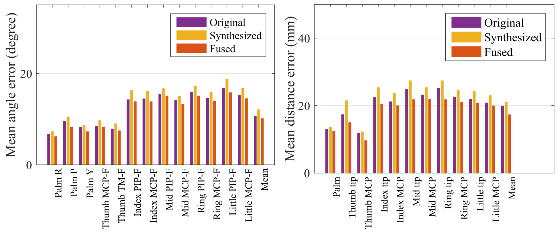
Quantitative evaluation We validate the proposed framework for hand pose estimation using our own synthetic dataset. Figure 7a shows the averaged angle error (in *degrees*) over all frames for each joint position. We observe that the error of the *Synthesized* (12.11) and *Original* (10.73) data is higher than that of the *Fused* (10.17) data all over the joint positions. It validates the rationale of the proposed data reproduction process. The consistent result is drawn in Figure 7b which presents the averaged distance error for each joint. Again, the use of the *Fused* images



(a)

(b)

Figure 6: Qualitative evaluations are conducted on (a) our synthetic dataset and (b) publicly available GUN-71 dataset. The first row shows the input depth image, and estimated hand skeletons are presented in the second row. The third row shows the reconstructed hand mesh model from skeleton estimation.



(a)

(b)

Figure 7: Quantitative evaluation on the overall robustness. (a) The individual mean joint angle error is used to compare the performance of the proposed method and baselines (in degrees). (b) Accuracy of hand pose estimation is examined as a function of the averaged joint distance (in mm) error.

outperforms the others over an entire range, validating our choice is overall more robust for pose estimation. In particular, the fact that the distance error of our palm position is less than average indicates our localization network well performs on the cluttered background in the presence of unseen objects.

Qualitative evaluation We conduct a qualitative evaluation of our approach using our synthetic dataset and publicly available GUN-71 dataset [19]. The top row of Figure 6a shows the input depth frames rendered using our 3D hand and object models. Note that the cluttered background was captured in-the-wild using a commercial depth camera. The second row shows the hand pose estimates using our framework. Finally, the reconstructed hand models are displayed in the third row. We observe that the proposed approach robustly estimates the valid and natural hand configurations against the severe object occlusions, various global orientations, and the cluttered background. Subsequently, the first row of Figure 6b shows the selected depth images of the GUN-71 dataset. Note that we use the first 33 classes of the GUN-71 dataset, which share the same grasp types with our dataset. The second and third row, respectively, shows the estimated poses and corresponding reconstruction based on our estimates. Figure 6 demonstrates that our approach performs robustly across input sources (*i.e.*, the data type and noise in acquired data)

7. Conclusion

We present a learning framework for hand pose estimation while interacting with an unknown object. Our main insight is that the shape of the object can be used to better represent the hand pose in the form of interactive grasps. By exploring their intimate relationship, more discriminative cues can be collaboratively derived from both perspectives. To generate a large database of the synthetic human grasps, we simulate 3D hand and CAD models. Using the dataset along with a ConvNet, we localize the center of the hand and object to create a pair of images. This pair is processed through the reproduction network to learn attributes of the synthetic images. We then classify the hand orientations and grasp type from the multi-channel network to reduce the search space for pose estimation. Finally, we compute the angle parameters from this subset. The evaluation results show that we achieve robust performance for both grasp classification and hand pose estimation. Future work will focus on varying attributes (*e.g.*, transparency) of the 3D object models and covering an entire camera viewpoint to reflect more realistic factors to our system.

Acknowledgements This work was partially supported by the NSF Award No.1637961 from NRI and 1329979 from CPS, as well as the Donald W. Feddersen Chaired Professorship from Purdue School of Mechanical Engineering. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

References

[1] L. Ballan, A. Taneja, J. Gall, L. Van Gool, and M. Pollefeys. Motion capture of hands in action using discriminative salient points. In *European Conference on Computer Vision*, pages 640–653. Springer, 2012. 2

[2] C. Choi, A. Sinha, J. Hee Choi, S. Jang, and K. Ramani. A collaborative filtering approach to real-time hand pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2336–2344, 2015. 1, 2, 3, 6

[3] T. Feix, J. Romero, H.-B. Schmiedmayer, A. M. Dollar, and D. Kragic. The grasp taxonomy of human grasp types. *IEEE Transactions on Human-Machine Systems*, 46(1):66–77, 2016. 1

[4] L. Ge, H. Liang, J. Yuan, and D. Thalmann. Robust 3d hand pose estimation in single depth images: From single-view cnn to multi-view cnns. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2

[5] C. Keskin, F. Kıraç, Y. E. Kara, and L. Akarun. Hand pose estimation and hand shape classification using multi-layered randomized decision forests. In *Computer Vision–ECCV 2012*, pages 852–863. Springer, 2012. 1, 2

[6] C. Keskin, F. Kıraç, Y. E. Kara, and L. Akarun. Real time hand pose estimation using depth sensors. In *Consumer Depth Cameras for Computer Vision*, pages 119–137. Springer, 2013. 1, 2

[7] P. Krejov, A. Gilbert, and R. Bowden. Guided optimisation through classification and regression for hand pose estimation. *Computer Vision and Image Understanding*, 155:124–138, 2017. 2

[8] N. Kyriazis and A. Argyros. Physically plausible 3d scene tracking: The single actor hypothesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9–16, 2013. 2

[9] N. Kyriazis and A. Argyros. Scalable 3d tracking of multiple interacting objects. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3430–3437. IEEE, 2014. 2

[10] E. Larsen, S. Gottschalk, M. C. Lin, and D. Manocha. Fast proximity queries with swept sphere volumes. Technical report, Technical Report TR99-018, Department of Computer Science, University of North Carolina, 1999. 3

[11] J. Liu, F. Feng, Y. C. Nakamura, and N. S. Pollard. A taxonomy of everyday grasps in action. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 573–580. IEEE, 2014. 1

[12] S. Melax, L. Keselman, and S. Orsten. Dynamics based 3d skeletal hand tracking. In *Proceedings of Graphics Interface 2013*, pages 63–70. Canadian Information Processing Society, 2013. 1, 2

[13] M. Oberweger, P. Wohlhart, and V. Lepetit. Training a Feedback Loop for Hand Pose Estimation. In *Proceedings of the International Conference on Computer Vision*, 2015. 5

[14] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Efficient model-based 3d tracking of hand articulations using kinect. In *BMVC*, volume 1, page 3, 2011. 1, 2

[15] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Full DOF tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *2011 International Conference on Computer Vision*, pages 2088–2095. IEEE, 2011. 1, 2

[16] P. Panteleris, N. Kyriazis, and A. A. Argyros. 3d tracking of human hands in interaction with unknown objects. In *BMVC*, pages 123–1, 2015. 2

[17] C. Qian, X. Sun, Y. Wei, X. Tang, and J. Sun. Realtime and robust hand tracking from depth. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1106–1113. IEEE, 2014. 1, 4

[18] G. Rogez, J. S. Supancic, and D. Ramanan. First-person pose recognition using egocentric workspaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4325–4333, 2015. 1, 2, 3

[19] G. Rogez, J. S. Supancic, and D. Ramanan. Understanding everyday hands in action from RGB-D images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3889–3897, 2015. 1, 2, 6, 7, 8

[20] J. Romero, H. Kjellström, and D. Kragic. Hands in action: real-time 3d reconstruction of hands in interaction with objects. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 458–463. IEEE, 2010. 1, 2

[21] T. Sharp, C. Keskin, D. Robertson, J. Taylor, J. Shotton, D. Kim, C. Rhemann, I. Leichter, A. Vinnikov, Y. Wei, et al. Accurate, robust, and flexible real-time hand tracking. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3633–3642. ACM, 2015. 1, 2, 3, 6

[22] A. Sinha, C. Choi, and K. Ramani. 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*, pages 4150–4158, 2016. 1, 2, 3, 4, 6

[23] S. Sridhar, F. Mueller, M. Zollhöfer, D. Casas, A. Oulasvirta, and C. Theobalt. Real-time joint tracking of a hand manipulating an object from RGB-D input. In *European Conference on Computer Vision*, pages 294–310. Springer, 2016. 1, 2, 7

[24] X. Sun, Y. Wei, S. Liang, X. Tang, and J. Sun. Cascaded hand pose regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 824–832, 2015. 1, 2, 6

[25] A. Tagliasacchi, M. Schroeder, A. Tkach, S. Bouaziz, M. Botsch, and M. Pauly. Robust articulated-ICP for realtime hand tracking. Technical report, 2015. 4

[26] D. Tang, H. J. Chang, A. Tejani, and T.-K. Kim. Latent regression forest: Structured estimation of 3d articulated hand posture. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 3786–3793. IEEE, 2014. 1, 2

[27] D. Tang, T.-H. Yu, and T.-K. Kim. Real-time articulated hand pose estimation using semi-supervised transductive regression forests. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 3224–3231. IEEE, 2013. 2

[28] A. Thayananthan, R. Navaratnam, B. Stenger, P. H. Torr, and R. Cipolla. *Multivariate relevance vector machines for tracking*. Springer, 2006. 2

[29] J. Tompson, M. Stein, Y. Lecun, and K. Perlin. Real-time continuous pose recovery of human hands using convolutional networks. *ACM Transactions on Graphics (TOG)*, 33(5):169, 2014. 1, 2, 4, 5

[30] D. Tzionas, L. Ballan, A. Srikantha, P. Aponte, M. Pollefeys, and J. Gall. Capturing hands in action using discriminative salient points and physics simulation. *International Journal of Computer Vision*, 118(2):172–193, 2016. 7

[31] Y. Wang, J. Min, J. Zhang, Y. Liu, F. Xu, Q. Dai, and J. Chai. Video-based hand manipulation capture through composite motion control. *ACM Transactions on Graphics (TOG)*, 32(4):43, 2013. 2

[32] C. Xu and L. Cheng. Efficient hand pose estimation from a single depth image. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 3456–3462. IEEE, 2013. 2