ContactPose: A Dataset of Grasps with Object Contact and Hand Pose Supplementary Material

Samarth Brahmbhatt[1][0000−0002−3732−8865], Chengcheng Tang[3], Christopher D. Twigg[3], Charles C. Kemp[1], and James Hays[1,2]

[1] Georgia Tech, Atlanta GA, USA {samarth.robo,hays}@gatech.edu, charlie.kemp@bme.gatech.edu
[2] Argo AI
[3] Facebook Reality Labs {chengcheng.tang,cdtwigg}@fb.com

Abstract. The supplementary material includes a discussion on contact capture, accuracy evaluation of the hand pose and contact ground truth, MANO hand mesh [13] fitting details, network architectures, and implementation details for the learning algorithms. It also includes examples of the RGB-D imagery present in the ContactPose dataset along with 3D hand joints projected into those images. Next, we present slices through the data in the form of 1) object- and intent-specific hand contact probabilities, and 2) 'use' vs. 'hand-off' contact maps and hand poses for all grasps of an object. Finally, we present the list of objects and their 'use' instructions, and describe the participants' hand information that is included in ContactPose.

1 Contact Capture Discussion

The process to convert thermal image pixels to contact values follows [4]. Raw thermal readings were converted to continuous contact values in $[0, 1]$ using a sigmoid that maps the warmest point to 0.95 and the coldest point to 0.05. These values non-linearly encode the temperature of the object, where $[0, 1]$ approximately corresponds to [room temperature, body temperature]. While most experiments used this continuous value, if a hard decision about the contact status of a point was desired, this was done by thresholding these processed values at 0.4.

2 MANO Fitting

This section provides details for the fitting procedure of the MANO [13] hand model to ContactPose data. Borrowing notation from [13], the MANO model is a mesh with vertices $M(\beta, \theta)$ parameterized by shape parameters β and pose parameters θ. The 3D joint locations of the posed mesh, denoted here by $J(\beta, \theta)$, are also a function of the shape and pose parameters. We modify the original

model by adding one joint at each fingertip, thus matching the format of joints J^* in ContactPose annotations.

MANO fitting is performed by optimizing the following objective function, which combines L2 distance of 3D joints and shape parameter regularization:

$$\beta^*, \theta^* = \arg min_{\beta,\theta} ||J(\beta, \theta) - J^*|| + \frac{1}{\sigma}||\beta|| \quad (1)$$

where σ is set to 10. It is optimized using the Dogleg [11] optimizer implemented in chumpy [2]. We initialized β and θ to $\mathbf{0}$ (mean shape and pose) after 6-DOF alignment of the wrist and 5 palm joints. Finally, the MANO model includes a PCA decomposition of 45 pose parameters to 6 parameters by default. We provide MANO fitting data with 10 and 15 pose components in the ContactPose dataset, but use the MANO models with 10 pose components in all our contact modeling experiments.

3 Dataset Accuracy

In this section, we cross-evaluate the accuracy of the hand pose and contact data included in ContactPose.

3.1 Contact Accuracy

We note that conduction is the principal mode of heat transfer in solid-to-solid contact [9]. Combined with the observation by Brahmbhatt *et al.* [4] that heat dissipation within the 3D printed objects is low over the time scales we employ to scan them, this indicates that conducted heat can accurately encode contact. Following [4], we measure the conducted heat with a thermal camera.

Agreement with MANO Hand Mesh: The average distance of contacted object points from their nearest hand point is 4.17 mm (10 MANO hand pose parameters) and 4.06 mm (15 MANO hand pose parameters).

Agreement with Pressure-based Contact: We also verified thermal contact maps against pressure images from a Sensel Morph planar pressure sensor [3,15]. After registering the thermal and pressure images, we thresholded the processed thermal image at values in $[0, 1]$ with an interval of 0.1. Any nonzero pixel in the pressure image is considered to be contacted. Binary contact agreement peaks at 95.4% at the threshold of 0.4 (Figure 1).

3.2 3D Hand Pose Accuracy

Following [6], this is measured through the discrepancy between 3D joints of the fitted MANO model and the ground truth 3D joints. Low-quality physically implausible ground truth can yield higher discrepancy, since the MANO model is not able to fit to physically implausible joint locations. Table 1 shows that ContactPose has significantly lower discrepancy than other recent datasets, even though it uses less than one-third MANO hand pose parameters. Table 2 shows statistics for hand-object penetration.

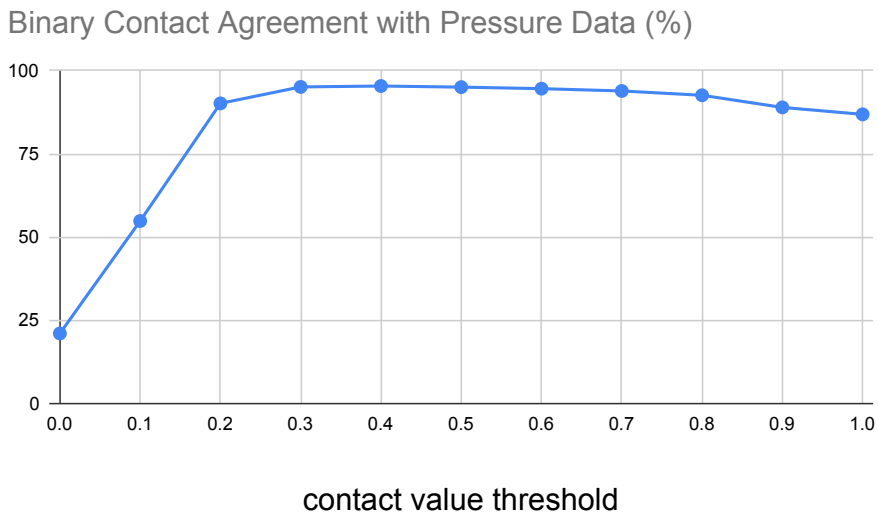

Fig. 1: Relation of contact value threshold to the binary contact agreement with pressure data from the Sensel Morph sensor. Agreement maximizes at the threshold value of 0.4.

Dataset	Avg. 3D Joint Error (mm)	AUC (%)
HO-3D [6]	7.7	79.0
FreiHand [16]	-	79.1
HANDS 2019 [1]	11.39	-
ContactPose (ours) – 10 pose params	7.65	84.54
ContactPose (ours) – 15 pose params	**6.68**	**86.49**

Table 1: Discrepancy between 3D joints of the fitted MANO model and the ground truth 3D joints. 3D joint error (lower is better) is averaged over all 21 joints. AUC (higher is better) is the area under the error threshold vs. percentage of correct keypoints (PCK) curve, where the maximum error threshold is set to 5 cm.

Dataset	Mean Penetration (mm)	Median Penetration (mm)	Penetration freq (%)
FPHA [5] (reported in [7])	11.0	-	-
ContactPose – 15 pose params	**2.02**	1.53	4.75

Table 2: Statistics for hand-object penetration showing the accuracy of ContactPose. Note that [7] report *joint* penetration for [5], while we report *surface* penetration.

4 Participants' Hand Information

We captured information about each ContactPose participant's hands in two ways: 1) contact map on a flat plate (example shown in Figure 2), and 2) RGB-D videos of the participants performing 7 hand gestures (shown in Figure 3). This can potentially be used to estimate the hand shape by fit embodied hand models (e.g. [13]).

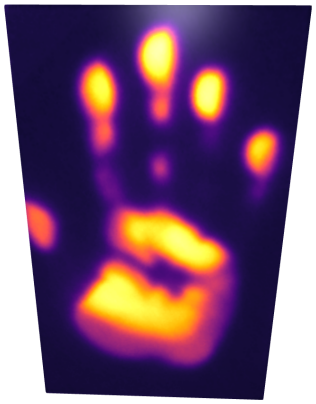

Fig. 2: Contact map of a participant's palm on a flat plate. Such palm contact maps for each participant are included in ContactPose.

5 Network Architectures

5.1 PointNet++

The PointNet++ architecture we use is similar to the pointcloud segmentation network from Qi et al [12], with modifications aiming to reduce the number of learnable parameters. Similarly to [12], we use $SA\left(s, r, [l_1, \ldots, l_d]\right)$ to indicate a Set Abstraction layer with a farthest point sampling ratio s, ball radius r (the pointcloud is normalized to lie in the $[-0.5, 0.5]$ cube) and d fully connected layers of size $l_i (i = 1 \ldots d)$. The global Set Abstraction layer is denoted without farthest point sampling ratio and ball radius. $FP\left(K, [l_1, \ldots, l_d]\right)$ indicates a Feature Propagation layer with K nearest neighbors and d fully connected layers of size $l_i (i = 1 \ldots d)$. $FC\left(S_{in}, S_{out}\right)$ indicates a fully connected layer of output size S_{out} applied separately to each point (which has S_{in}-dimensional features). Each fully connected layer in the Set Abstraction and Feature Propagation layers

Fig. 3: Pre-defined hand gestures performed by each participant. RGB-D videos from 3 Kinects of each participant performing these gestures are included in ContactPose.

is followed by ReLU and batch-norm layers. Our network architecture is:

$$SA\left(0.2, 0.1, [F, 64, 128]\right) - SA\left(0.25, 0.2, [128, 128, 256]\right) - SA\left([256, 512, 1024]\right) - FP(1, [1024 + 256, 256, 256]) - FP(3, [256 + 128, 256, 128]) - FP(3, [128 + F, 128, 128]) - FC(128, 128) - FC(128, 10)$$

where F is the number of input features and the final layer outputs scores for the 10 contact value classes.

5.2 Image Encoder-Decoder

We take inspiration from U-Net [14] and design the light-weight network shown in 4 that extracts dense features from RGB images. The global average pooling layer is intended to capture information about the entire hand and object.

6 Training and Evaluation Details

All models are trained using PyTorch [10] and the Adam optimizer [8] (base learning rate $\in \{5 \times 10^{-4}, 1 \times 10^{-3}, 5 \times 10^{-3}\}$, momentum of 0.9, weight decay

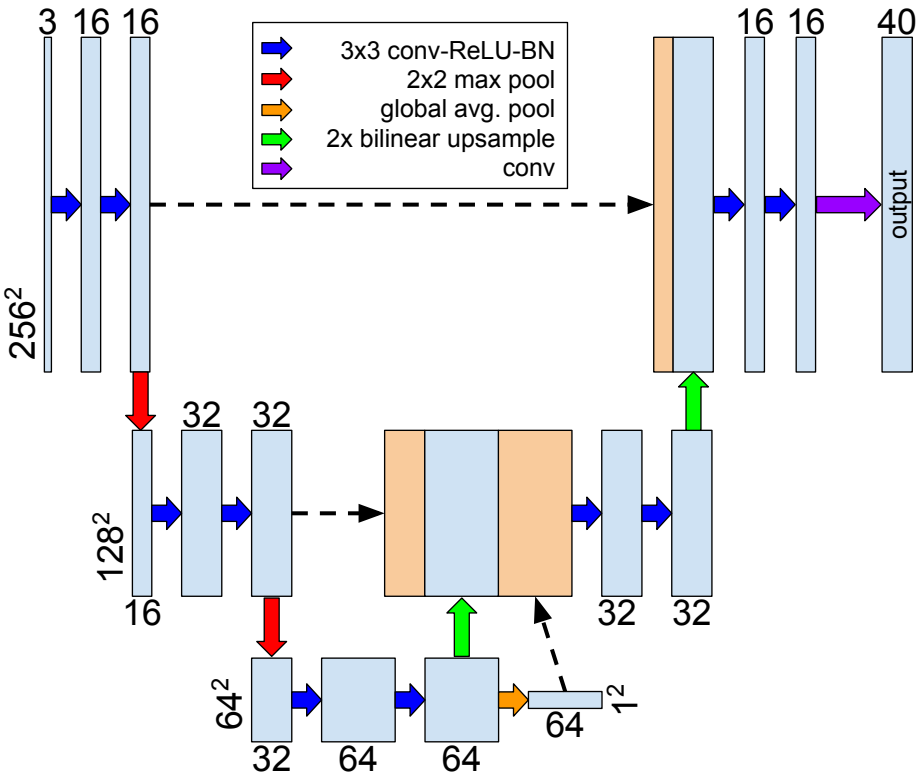

Fig. 4: Architecture for the image encoder-decoder (Figure 10 in main paper). Horizontal numbers indicate number of channels, and vertical numbers indicate spatial dimensions.

of $5e-4$, and a batch size of 25). Both point-clouds and voxel-grids are rotated around their 'up'-axis at regularly spaced 30° intervals. These rotations are considered separate data points during training, and their predictions are averaged during evaluation.

For image-based contact prediction, ContactPose has approximately 300 RGB-D frames ($\times$ 3 Kinects) for each grasp, but temporally nearby frames are highly correlated because of the high frame rate. Hence, we include equally spaced 50 frames from each grasp in the training set. Evaluation is performed over equally spaced 12 frames from this set of 50 frames.

7 List of Objects

Table 3 shows a list of all 25 objects in ContactPose, along with information about the which of these objects are included in the two functional grasping categories, and the specific 'use' instructions.

Object	**handoff**	**use**	**use instruction**
apple	✓	✓	eat
banana	✓	✓	peel
binoculars	✓	✓	see through
bowl	✓	✓	drink from
camera	✓	✓	take picture
cell phone	✓	✓	talk on
cup	✓	✓	drink from
door knob		✓	twist to open door
eyeglasses	✓	✓	wear
flashlight	✓	✓	turn on
hammer	✓	✓	hit a nail
headphones	✓	✓	wear
knife	✓	✓	cut
light bulb	✓	✓	screw in a socket
mouse	✓	✓	use to point and click
mug	✓	✓	drink from
pan	✓	✓	cook in
PS controller	✓	✓	play a game with
scissors	✓	✓	cut with
stapler	✓	✓	staple
toothbrush	✓	✓	brush teeth
toothpaste	✓	✓	squeeze out toothpaste
Utah teapot	✓	✓	pour tea from
water bottle	✓	✓	open
wine glass	✓	✓	drink wine from
Total	**24**	**25**	

Table 3: List of objects in ContactPose and specific 'use' instructions

8 Example Data from ContactPose

RGB-D Images with Projected Hand Pose: Figure 5 shows example RGB and depth images (256×256 crops centered around the object) for all objects, along with projected 3D joints.
Hand Contact Probabilities: Figure 6 shows (phalange-level) hand-part contact probabilities (similar to Figure 7(b) in the main paper) for all objects, averaged separately over 'use' and 'hand-off' grasps. Many objects that elicit significantly different 'use' and 'hand-off' contact patterns, e.g. cellphone, flashlight, hammer, knife, mouse, pan, PS controller, stapler, toothbrush, and toothpaste. The 'use' grasps for banana and water-bottle have different contact patterns on the left and right hand, because many participants use their non-dominant hand to hold them firmly in an enveloping grasp and the dominant hand to peel and open the cap, respectively.
Grasps: To further demonstrate the scale and diversity of ContactPose data, we present a slice of the data. Figures 7 and 8 show all the 'use' and 'hand-off' grasps (contact map and hand pose) for one object (PS-controller), respectively. Note the significant influence of intent on grasps, and also the intra-intent diversity of grasps.

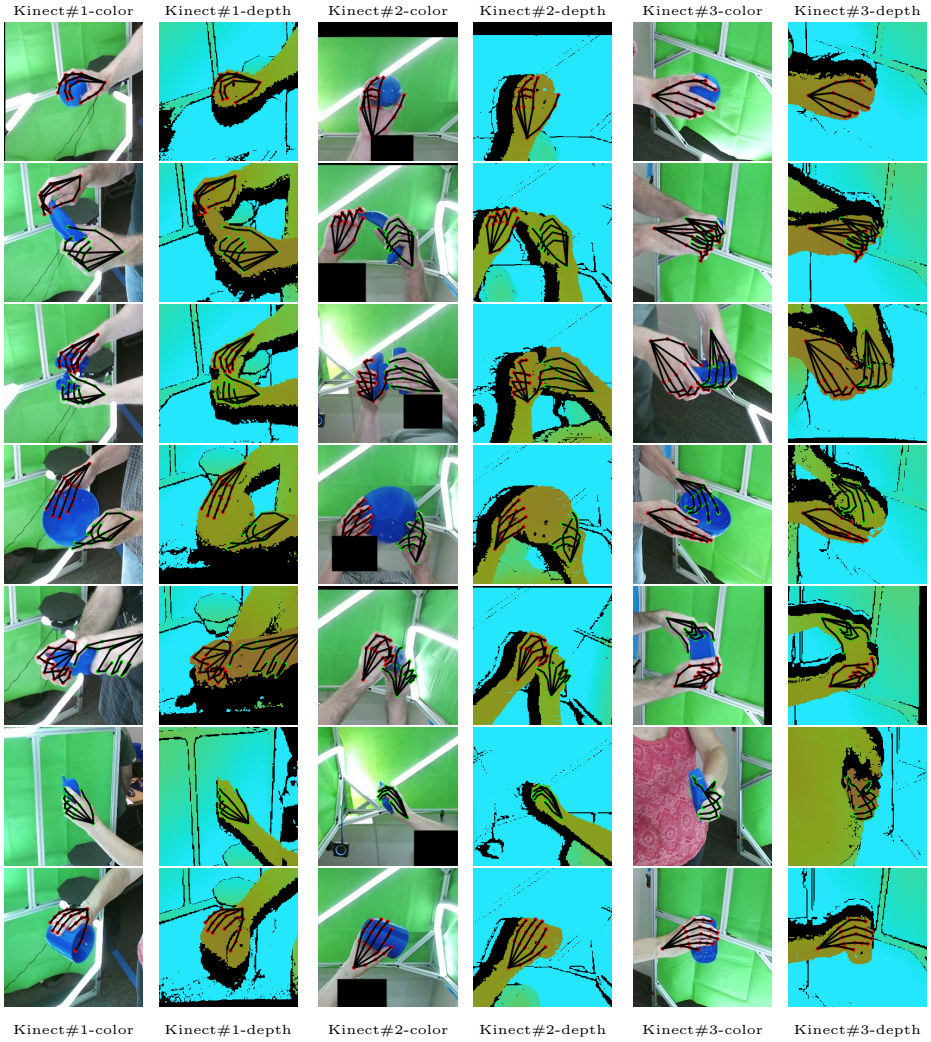

Fig. 5: Example RGB and depth images from ContactPose ('use' intention), with 3D joint locations projected into the images. Left hand joints are **green**, right hand joints are **red** (continued below).

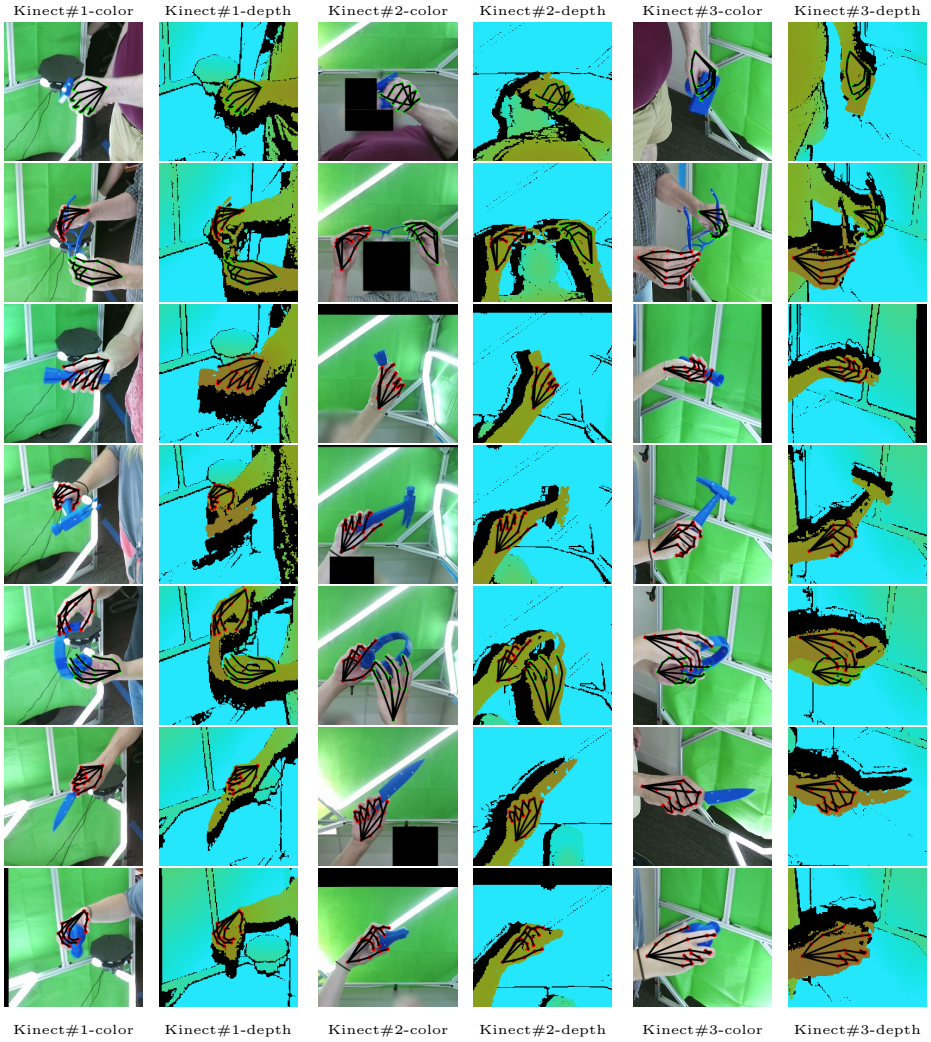

Fig. 5: Example RGB and depth images from ContactPose ('use' intention), with 3D joint locations projected into the images. Left hand joints are **green**, right hand joints are **red** (continued below).

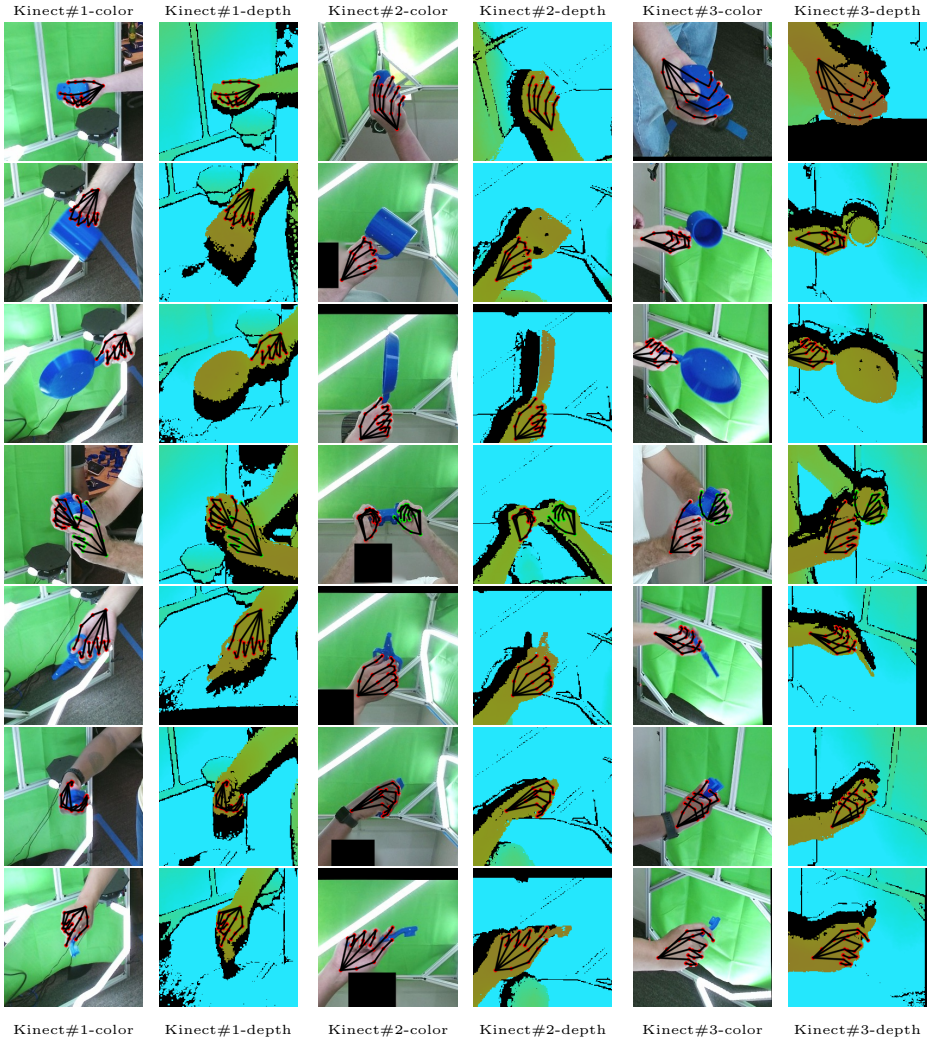

Fig. 5: Example RGB and depth images from ContactPose ('use' intention), with 3D joint locations projected into the images. Left hand joints are **green**, right hand joints are **red** (continued below).

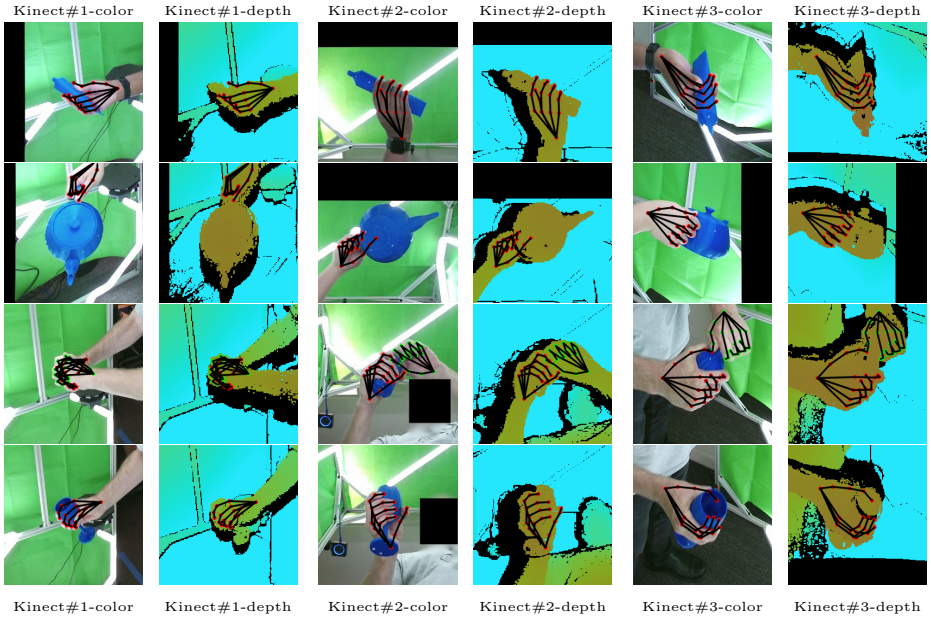

Fig. 5: Example RGB and depth images from ContactPose ('use' intention), with 3D joint locations projected into the images. Left hand joints are **green**, right hand joints are **red**.

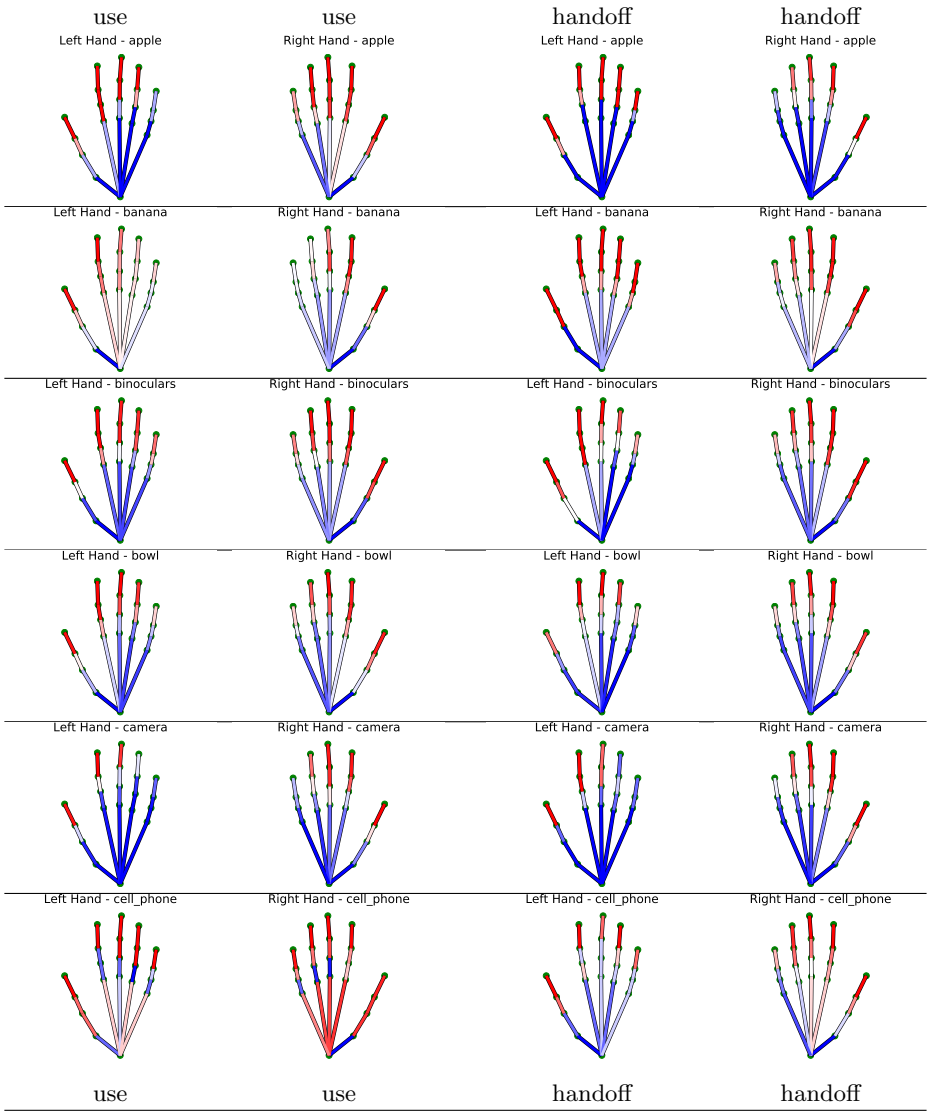

Fig. 6: Hand-part contact probabilities for objects in ContactPose (similarly to Figure 5 in the main paper, **red** indicates high probability and **blue** indicates low probability) (continued below).

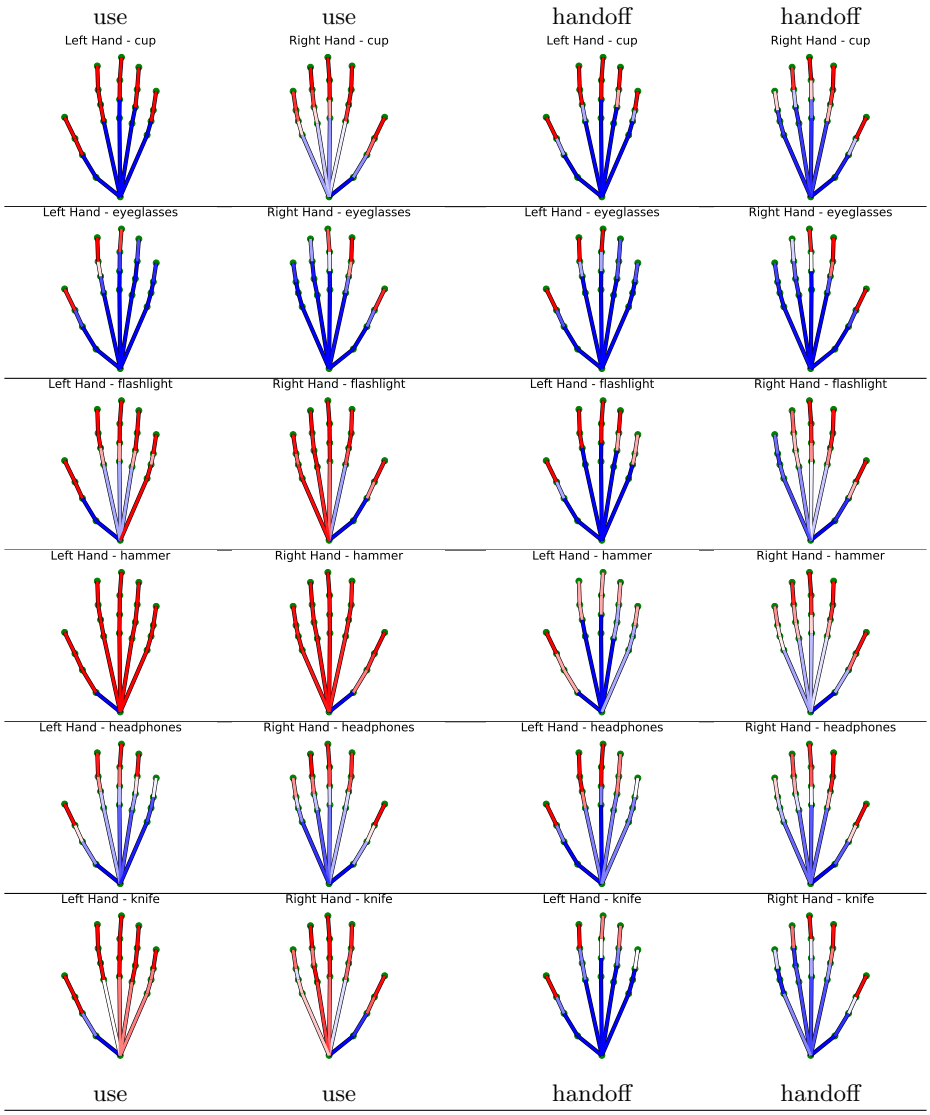

Fig. 6: Hand-part contact probabilities for objects in ContactPose (similarly to Figure 5 in the main paper, **red** indicates high probability and **blue** indicates low probability) (continued below).

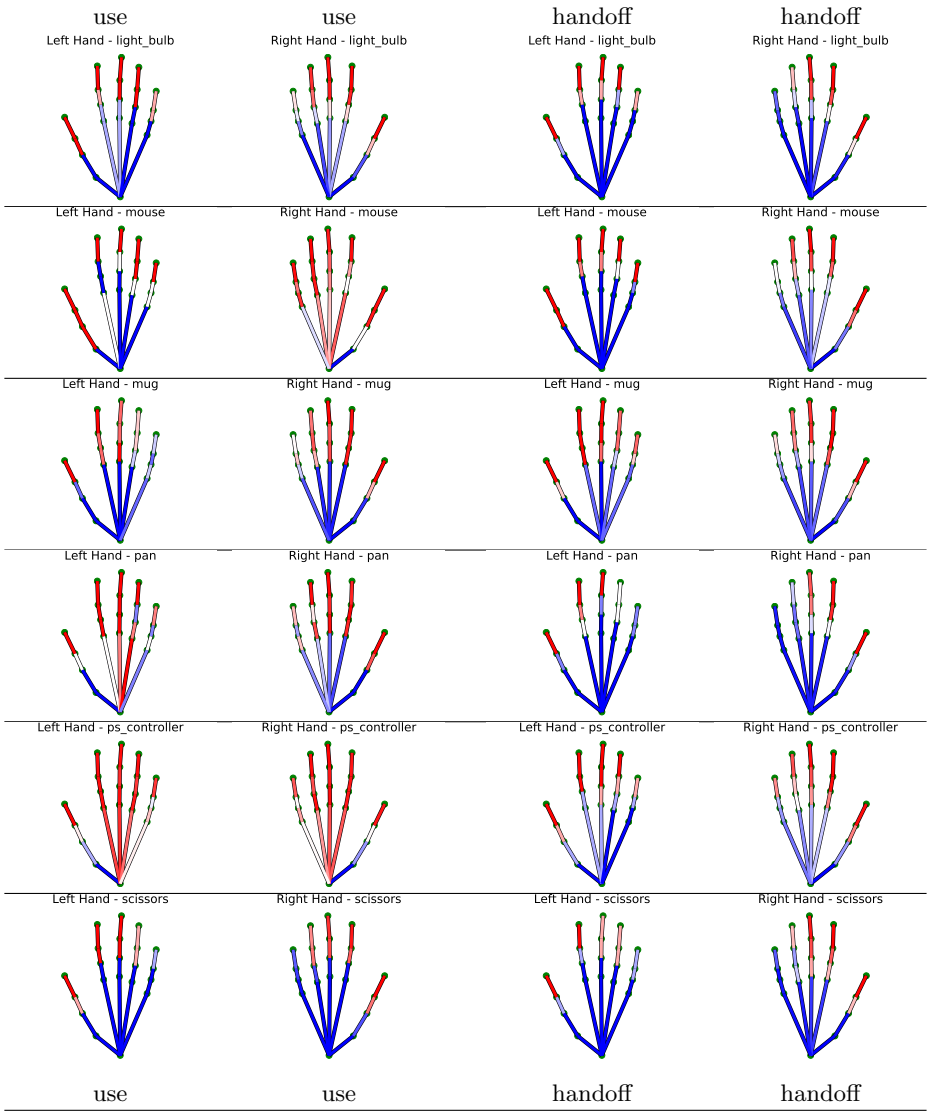

Fig. 6: Hand-part contact probabilities for objects in ContactPose (similarly to Figure 5 in the main paper, **red** indicates high probability and **blue** indicates low probability) (continued below).

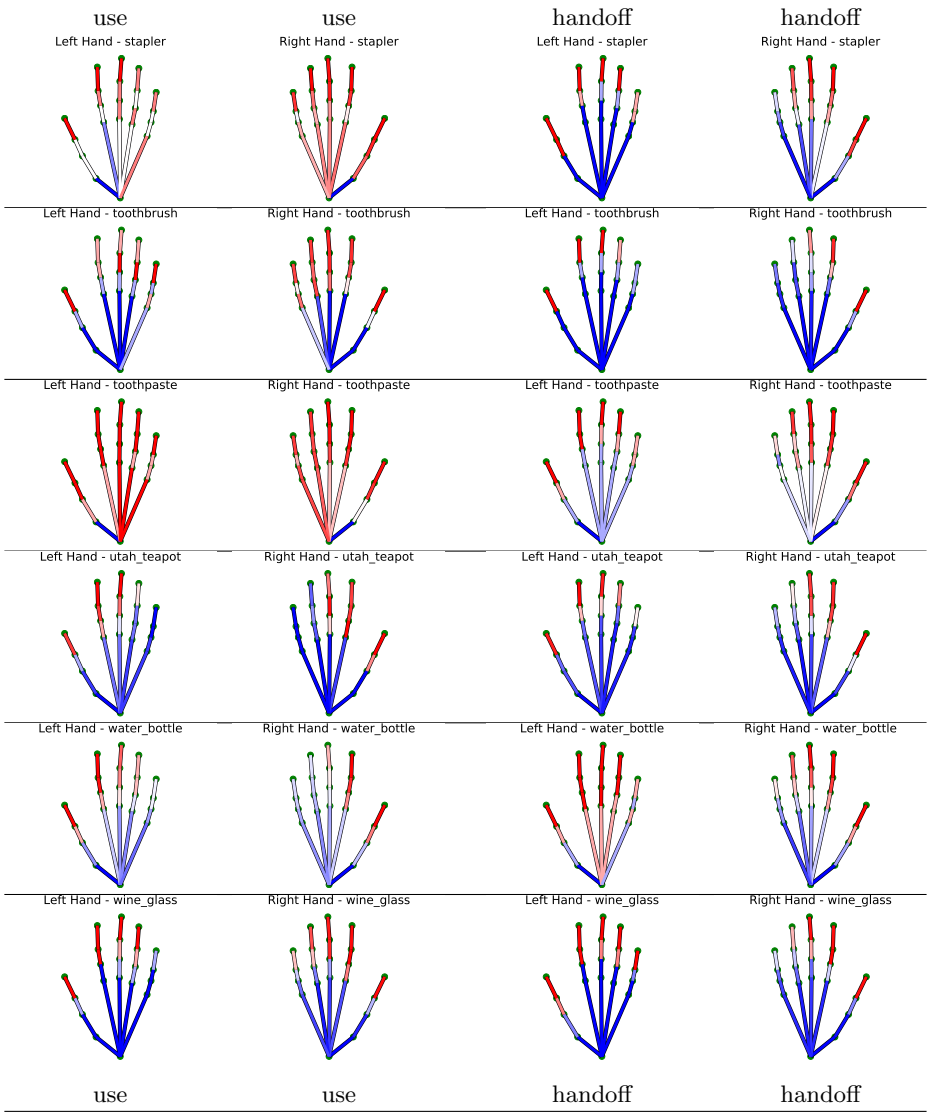

Fig. 6: Hand-part contact probabilities for objects in ContactPose (similarly to Figure 5 in the main paper, **red** indicates high probability and **blue** indicates low probability). Data is grouped by left or right hand and by 'use' or 'hand-off' intent.

Fig. 7: A slice through ContactPose: All PS-controller 'use' grasps (2 views per grasp) (continued below).

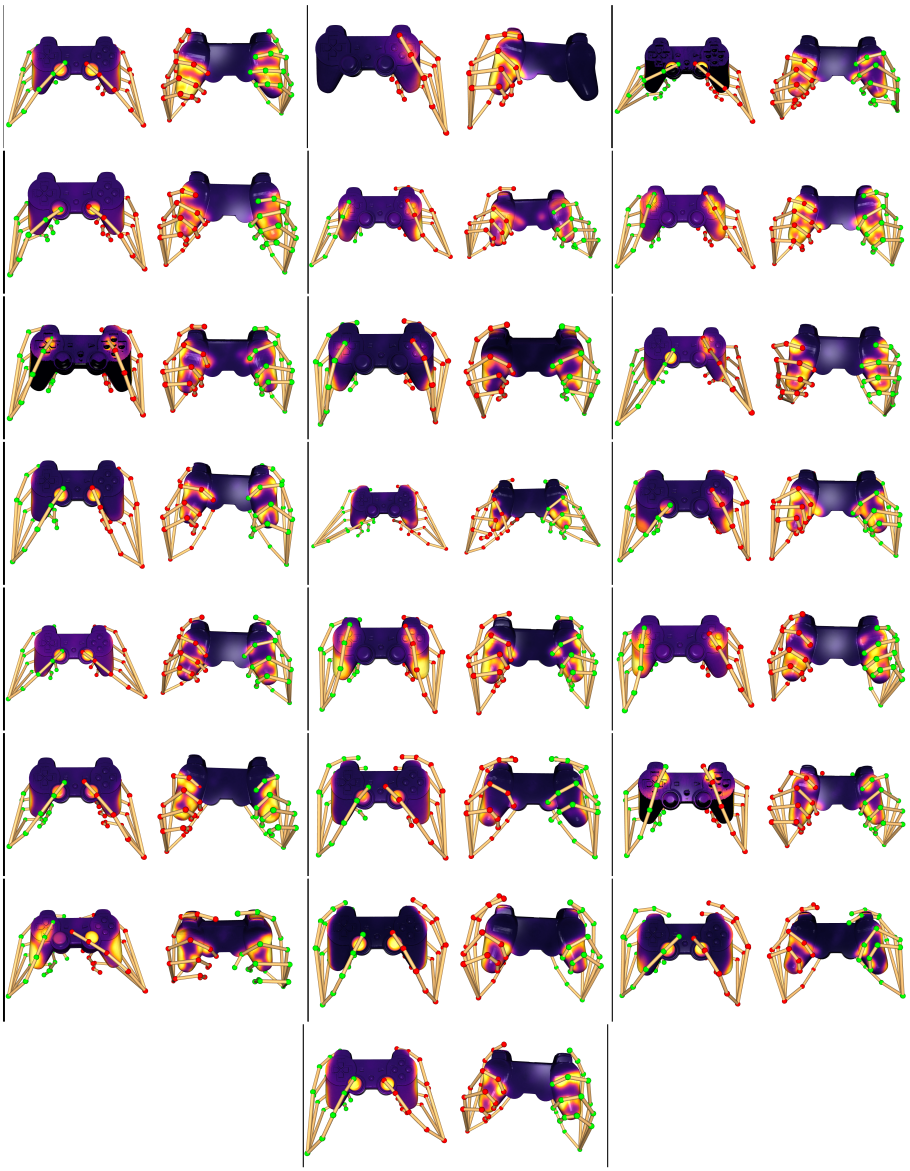

Fig. 7: A slice through ContactPose: All PS-controller 'use' grasps (2 views per grasp).

Fig. 8: A slice through ContactPose: All PS-controller 'hand-off' grasps (2 views per grasp) (continued below).

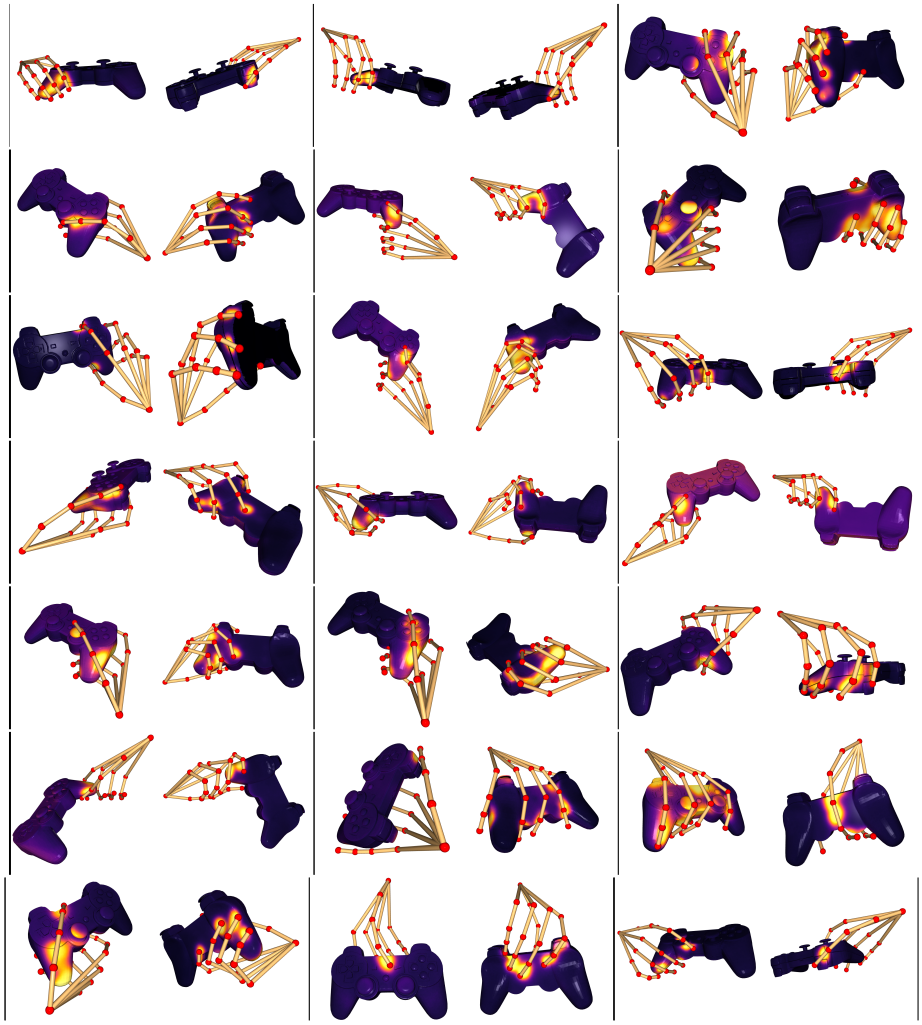

Fig. 8: A slice through ContactPose: All PS-controller 'hand-off' grasps (2 views per grasp).

References

1. 5th International Workshop on Observing and Understanding Hands in Action. https://sites.google.com/view/hands2019/challenge#h.p_adfpp7VAhgAL, accessed: 2020-03-12 3
2. chumpy: Autodifferentiation tool for Python. https://github.com/mattloper/chumpy, accessed: 2020-03-12 2
3. Sensel morph. https://sensel.com/pages/the-sensel-morph, accessed 2020-07-07 2
4. Brahmbhatt, S., Ham, C., Kemp, C.C., Hays, J.: ContactDB: Analyzing and predicting grasp contact via thermal imaging. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019) 1, 2
5. 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 Computer Vision and Pattern Recognition (CVPR) (2018) 3
6. Hampali, S., Rad, M., Oberweger, M., Lepetit, V.: Honnotate: A method for 3d annotation of hand and object poses. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020) 2, 3
7. Hasson, Y., Varol, G., Tzionas, D., Kalevatykh, I., Black, M.J., Laptev, I., Schmid, C.: Learning joint reconstruction of hands and manipulated objects. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 11807–11816 (2019) 3
8. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 5
9. Lienhard, IV, J.H., Lienhard, V, J.H.: A Heat Transfer Textbook. Phlogiston Press, Cambridge, MA, 5th edn. (Aug 2019), http://ahtt.mit.edu, version 5.00 2
10. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in PyTorch. In: NIPS Autodiff Workshop (2017) 5
11. Powell, M.J.: A new algorithm for unconstrained optimization. In: Nonlinear programming, pp. 31–65. Elsevier (1970) 2
12. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: Advances in neural information processing systems. pp. 5099–5108 (2017) 4
13. Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. ACM Transactions on Graphics (TOG) **36**(6), 245 (2017) 1, 4
14. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015) 5
15. Rosenberg, I.D., Zarraga, J.A.: System for detecting and confirming a touch input. US Patent US20170336891A1 (2017) 2
16. 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: The IEEE International Conference on Computer Vision (ICCV) (October 2019) 3