GRAB: A Dataset of Whole-Body Human Grasping of Objects *Supplemental Material*

Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas

Max Planck Institute for Intelligent Systems, Tübingen, Germany {otaheri,nghorbani,black,dtzionas}@tuebingen.mpg.de

https://grab.is.tue.mpg.de

S.1 Supplementary Video

The video on our website presents:

- a narrated overview of our method,
- a wide variety of GRAB sequences (3D moving meshes),
- GrabNet predictions for unseen objects from several viewpoints and
- GrabNet failure cases.

S.2 GRAB Dataset Content

Our GRAB dataset is available for research purposes on our website. The website contains (at least) the material listed below:

- Our modified version of the object meshes of [2].
- Our marker locations on each object mesh.
- Body shape templates for our subjects.
- Pose parameters for our subjects and objects.
- Code to reproduce the interacting meshes, as seen in our video.
- Per-vertex contact annotations on meshes (body and object) for each frame.
- Vicon MoCap files (labeled marker positions, incl. on the floor and table).

S.3 Why MoCap Instead of 3D Scan Sequences?

For accurate human shapes, we capture a dense 3D scan for each subject to which we fit a personalized 3D SMPL-X template mesh. However, 3D scanning does not scale up for capturing human-object interaction sequences. This would produce huge amounts of data, the processing of which would be a major undertaking. Moreover, object tracking under occlusions would be still very challenging, as finding scan-to-model correspondences is a hard ill-posed problem. Instead, with MoCap a minimum of 3 marker observations is enough for reliable object pose estimation. The placement of many small markers on the objects means that we can always estimate object pose. Using MoSh++ for the body, given a ground truth body shape, produces accurate meshes that are on par with 3D scanning but much more practical to capture. We follow therefore this practical and scalable approach; we use a high-end optical MoCap system (Sec. 3.1) and fit full 3D meshes to MoCap markers (Sec. 3.2) for both the human and the object.

S.4 Why not Capture RGB Images?

Capturing *accurate* human-object interactions while also capturing *natural* RGB images is very challenging. Some recent datasets [4, 5] capture hand-only interactions with objects and include RGB images, but the images capture only the hand and not the whole body [4, 5] and are not fully natural due to visible instrumentation on the hand [4]. Please note that this latter point is fundamental. Currently one must choose between accurate grasping, which requires instrumentation, or natural images, which reduces the accuracy of ground truth.

Both methods [4, 5] suffer from severe hand-object inter-penetrations. Garcia-Hernando et al. [4] originally reconstruct a hand skeleton interacting with 4 object meshes, and their method was reported to have an average *skeleton* penetration depth of 11.0 ± 8.9 mm (see Sec. 5.2 of [6]). Similarly, we compute the *surface* penetration between MANO and the 3D object meshes for [5] and find the mean to be 4.36 ± 0.94 mm. Although the hand inter-penetration of [5] is not as severe as [4], it suffers from not having realistic contact with objects. In Fig. S.1 we compare the contact "heatmaps" for [5] (left) and GRAB (right). Note that for [5], the thumb and all fingertips are rarely in contact, whereas for GRAB they are frequently in contact. The latter is much more realistic given the central role of the thumb and fingertips in object manipulation. This points to an important technical problem, without lowering the value of these works, as they focus on a challenging application.

We conclude that state-of-the-art interaction methods, that also capture RGB images, suffer from intense occlusions and penetrations along with non-realistic contact between the hand and the objects. Such data is not good to learn an accurate data-driven model of 3D interactions. In contrast, our "use" grasps have only 3.25 ± 0.68 mm average *surface* penetration, which is significantly lower than [4, 5], and realistic contact between the body and objects, while containing more challenging scenarios, namely dexterous in-hand manipulation and capture of the whole body instead of only the hand.

This is attributed to our precision-focused setup, that increases accuracy on the expense of not capturing RGB images, due to the uniform and artificial texture of the MoCap body suit and the 3D printed objects. We believe that this is a sensible trade-off; one can use our accurate 3D mesh reconstructions to learn a model of 3D interactions, and use it as a prior in future work to improve methods like [4, 5] for the hand or the whole body.

3



Fig. S.1: Contact "heatmaps" for HO-3D [5] (left) and GRAB (right), for the right hand. The hotter the color, the more frequently that hand part is in contact with objects. During grasping and manipulation, the thumb finger and all fingertips play a central role. This is evident with GRAB but not with [5].

S.5 Penetration Plots

In the main paper we describe the observed human-object mesh inter-penetration. Due to space constraints, we report here the corresponding plots in Fig. S.2. We evaluate the degree of penetration for "use" sequences, that pose the most realistic occlusions and capture challenges. "Use" grasps have 3.25 ± 0.68 mm average penetration, which effectively corresponds to the missing soft-tissue deformation. Please note that there is no model of the human body with articulated fingers and face that captures such soft-tissue deformation with contact. In addition, 67% of "use" grasps have ≤ 3.5 mm penetration, $86\% \leq 4.0$ mm, $96\% \leq 4.5$ mm and 99.9% has ≤ 5.8 mm.

S.6 Protocol Details

Here we provide details that were not crucial for the main manuscript (Sec 3.4). We capture motions with 4 intents: "use", "pass", "lift", and "off-hand pass". For each sequence we randomize the object position and pose on a resting table, the height of which is also randomized between 75 cm and 120 cm to increase motion variance. We capture the following intents:

"Use": For the objects that have a clear everyday use (e.g. drinking from cup), we ask the subject to naturally use them. In case of multiple use cases (e.g. digital/analog photo camera) we capture multiple sequences. For objects without a clear use (e.g. cylinder) the subject has to grasp them and inspect them.

"**Pass**": The subject is asked to pass the object to a predefined direction, that is randomized (e.g. bottom-left, top-right, etc), to increase motion variance.

"Lift": The subject is asked to grasp the object, lift it stably in any natural way



Fig. S.2: Penetration plots for "use" grasps. For each frame we store the max (red), mean (green) and median (blue) vertex penetration. (Left): percentage of frames (Y axis) below a varying penetration error (X axis). (Right): bar-plot for number of frames (Y axis) with a specific (quantized for binning) penetration (X axis). The mean penetration is 3.25 ± 0.68 mm.

they can imagine, then leave it on the table in any random pose, and repeated this several times with re-grasping. This increases grasp variance, by encouraging the exploration of contact configurations and relative hand-object orientations. "Off-Hand pass": As a form of bimanual manipulation, the subject grasps the object with the off hand, passes it to the dominant hand, and uses it (see "use").

We capture MoCap markers placed on the body, face and fingers, as well as on the object (Sec 3.1 in paper). Additionally, we capture markers attached to the floor and the table, for potential future use. All subjects gave informed consent to share their motion data for research purposes.

S.7 Computing Contact

Here we provide some additional intuition to Sec. 3.3 and Fig. 4 (right). In particular we explain how we deal with noise in the reconstructed moving meshes to produce clean contact data.

Figure S.3 (left): Consider the illustrated example of a 3D cup. Its mesh thickness is thin, i.e. it has an outer and inner surface that are different, yet close to each other. In Fig. S.3 (top-left) the thumb and index fingers of a grasping hand penetrate both the outer and inner surface. This is due to noise, fitting errors, and because existing models do not model contact-dependent skin deformations.

For these examples the actual contact area is the one on the outer object surface. To annotate only this, following Sec. 3.3, we first compute all colliding triangles and cluster them in connected "rings" (Fig. S.3, top-left). For each "ring" we compute the corresponding penetrating hand areas (Fig. S.3, bottom). The hand areas that contact the inner surface are a subset of the ones that contact the outer one. Then, we remove (big red circles in Fig. S.3 bottom) the

5



Fig. S.3: (Left): Annotating contact areas for a hand grasping a cup: (top-left) "rings" of colliding triangles, color-coded for each finger, (bottom) penetrating hand areas that correspond to each "ring"; the ones corresponding to the green and purple vertices (red circled hand parts) penetrate the inner cup surface and are ignored, (top-right) the final filtered "rings" and the enclosed vertices are annotated as contact areas. The contact labels are binary; color is used here only for visualization purposes. (**Right**): Contact labels can though be more fine-grained, e.g. using the contacting hand parts or hand vertices. Here we see an example of the former case. (top) Each color represents a contact area caused by a different hand part. (bottom) Contact areas are shown also on the object and unposed hand for clarity. Note that the size of contact areas is expanded for illustration purposes.

purple and green groups, we keep only the remaining "rings", and annotate the vertices enclosed by them as contact vertices (Fig. S.3, top-right).

Figure S.3 (right): The above procedure gives binary contact annotations ("contact" on "not in contact"). Contact labels, however, can be more finegrained, e.g. with the label of the corresponding hand part (Fig. S.3, right), or even with the point on the 3D hand surface (Fig. S.4). For the former example, we find the object vertices that are in contact, and for each one we find the closest SMPL-X/MANO bone, and assign its ID as the contact label.

S.8 Adapting MoSh++

We adapt MoSh++ [9] (Sec. 3.2) for capturing the whole body (including the hands and face). The human and object are tracked independently and on a per-frame basis, for simplicity. We make two small changes to MoSh++. First we use the ground-truth body shape, obtained from a 3D scan. Consequently, we do not use MoSh++ to estimate body shape. Second, we extend MoSh++ to estimate the parameters of the SMPL-X body model. This means extending it to capture facial pose and expression parameters. Additionally, we estimate the rigid 6 DoF pose of the objects using their known shape and the detected markers.



Fig. S.4: (Right): Fine-grained contact labels. In contrast to the binary contact labels of Fig. 4 (left) and Fig. 9 of the main manuscript, and the part-based contact labels of Fig. S.3, here we show an example of much more fine-grained labels. (left) A 3D hand-object grasp configuration. (middle) The object alone. (right) The hand in canonical pose. We highlight different points on the 3D surface of the inner hand with color gradients. The contact between the hand and object define surface correspondences between them (shown as lines).

To adapt MoSh++ to capture faces, we need to tune the parameters of the model. For this [9] follows a data-driven approach; they capture the SSM dataset with an optical MoCap system synchronized with a 3D body scanner, and use the scans for computing a reconstruction quality metric. However, SSM has markers only on the main body, while also the fingers of the scans are very noisy.

Capturing such a dataset, with clear scan regions for both the body, the face and all fingers, as well as synced MoCap for them, is too challenging. Instead, we follow a more practical approach, and create a synthetic dataset by animating SMPL-X and generating virtual markers on the moving meshes. To bridge the domain gap, we simulate noise for marker position and visibility; we randomly add 3D Gaussian noise with 1 mm variance in marker positions, as in [9], and randomly drop up to 5 markers per frame.

Unfortunately, there is no existing dataset with rich SMPL-X sequences. However, its model formulation is compatible to SMPL [8] for the body, FLAME [7] for the face, and MANO [13] for the hands. Therefore, we resort to datasets specific to each part to animate the body, face and hands. For the *body*, we employ DFAUST [1] that captures 10 subjects performing 10 sequences each. We split the subjects into 6 for training and 4 for a withheld test set. We compute personalized SMPL-X mesh templates by registering the model to one scan per person as in [13, 10], and pose their body according to the registrations of DFAUST. For the hand, we employ the hand-only MANO model registrations of [13]. From the 1554 hand poses we hold out 155 for the test set and use the rest for training. We then add hand motion to each body sequence by randomly choosing 15 hand poses and interpolating between them. For the *face*, we employ sequences of FLAME parameters from [3, 12]; the latter covers extreme facial expressions, while the former has everyday speaking expressions. We randomly choose 100 sequences from each dataset, splitting them into 60 for the training set and 40 for the withheld test set.

Table S.1: Evaluation of MoSh++ on the synthetic dataset. We compare the vanilla [9] to our adapted version. For the first stage of MoSh++, Stage-I, we report the distance of the latent marker placement compared to ground-truth marker locations, and for the second stage of MoSh++, Stage-II, we report the average vertex-to-vertex error between estimated and ground-truth meshes.

Mosh++	MoSh Stage-I		MoSh Stage-II		
version	mean \pm std	median	mean \pm std	median	
Vanila [9]	4.76 ± 1.03	4.55	5.59 ± 1.86	5.28	В
Our adapted	$\textbf{3.09} \pm \textbf{0.55}$	2.80	$\textbf{4.86} \pm \textbf{1.83}$	4.48	E

We use this dataset to set the weights following the approach in [9]. Table S.1 compares a standard version of [9] with our adapted version on the synthetic test set for both stages of MoSh++. For the first stage of MoSh++ (Stage-I) we report the distance of the latent marker placement compared to ground-truth marker locations in mm. For this stage we start from random marker placement guesses in the 1-ring neighborhood of the ground-truth locations. We repeat this three times with different random seeds for selecting 12 frames of MoSh++; see [9]. For the second stage (Stage-II), we use the optimized latent marker placements resulting from each random seed of the Stage-I and report the average vertex-to-vertex error between estimated and ground-truth meshes in mm. In each stage we choose the wights that minimizes the reported error. Our adapted version shows a clear improvement, by fitting the whole body, hands and face, with weights λ tuned on our synthetic dataset. In contrast [9] tunes only the body weights on their SSM dataset, it fits the hands with empirical weights, and does not fit the face.

S.9 GrabNet

To show that GRAB can be useful for machine learning applications, we train on it a generative model to generate grasping poses for a 3D object, that we call GrabNet. For this example application we focus only on right-hand grasps for simplicity, but GRAB provides much richer data than this.

S.9.1 Data Preparation

We use only right hand data due to the large size of GRAB, but left hand poses could also be mirrored to appear as right ones for data augmentation. In order to select right-hand frames for training GrabNet we use the following rules. (i) The right hand should be in contact. (ii) The left hand should not have any contact. (iii) The object's vertical position should be at least 5 mm different from its initial one (i.e. it should be lifted from the resting table). (iv) The right thumb and at least one more finger should be in contact. (v) A finger is considered a contacting finger, when it is in contact with at least 50 object vertices. With 8 O. Taheri et al.

these filters we make sure that we have only stable grasps with which to train GrabNet.

To model arbitrary shapes, we use the basis point set [11] representation BPS_o for all our objects. For computational efficiency, we precompute BPS_o and load it from memory during training. We sample basis points in a sphere of 150 mm radius, that is big enough to cover our centroid-centered objects. We empirically found 4096 basis points to be enough. We then compute the distances from the basis points to our object meshes.

Out of our 51 objects, borrowed from [2], we hold out 4 objects for the validation set ("apple", "toothbrush", "elephant" and "hand"), 6 objects for the test set ("mug", "wineglass", "camera", "binoculars", "frying pan" and "toothpaste"), and use the remaining 41 objects for the training set.

The training, validation and test splits contain roughly 320k, 31k and 65k data points, correspondingly.

To prepare the training data for RefineNet we add Gaussian noise to the Ground Truth MANO parameters of the selected data for GrabNet. Since the perturbation need to be minimal we empirically find $\mathcal{N}(\mu = 0, \sigma^2 = 0.2), \mathcal{N}(\mu = 0, \sigma^2 = 0.004)$, and $\mathcal{N}(\mu = 0, \sigma^2 = 0.05)$ for MANO finger joints rotation, root rotation, and translation respectively.

S.9.2 Results: Success and Failure Cases

Figures S.6, S.7 and S.8 provide a wide variety of qualitative GrabNet results. More specifically, they show 10 different grasps (rows) generated for 6 unseen objects (columns). The three figures show three different viewpoints (one view per figure) for the same grasp of the 10×6 grid. We see that most grasps look natural and plausible, as GrabNet is learned from high-quality GRAB captures. More results with a rotating viewpoint are shown in the video on our website.

GrabNet can still generate some failure cases. These are mostly cases of penetrating fingers; there are not many cases of contacting fingers that fly away from the object. Penetrations are observed mostly for objects with thin parts (cup handle, wine glass, bowl). We found the frying pan to be the most challenging object, due to its comparably big size along with its thin surface walls and handle. This might be due to the sparse BPS_o representation for 3D object shapes capturing mostly their bigger parts. Furthermore, at the moment we use a penetration and a contact term in the training loss of GrabNet as soft constraints, since here we focus on a data-driven method. One could add an optimization stage to refine the regressed grasp with hard penetration and contact constraints.

The results show the value of GRAB for training data-driven models, but also point to room for improvement for GrabNet's modeling and training scheme.

S.9.3 GrabNet Implementation Details

The architecture for GrabNet is shown in Fig. S.5. For CoarseNet, we concatenate the object BPS_o representation with MANO hand parameters as input to the

encoder, and also concatenate it with the latent code as the input (condition) to the decoder. The outputs of the decoder are MANO translation ($\gamma \in R^3$) and joint angles ($\theta \in R^{96}$) in the continuous 6-dimensional representation of [14].

Using our validation set, we found out that 16 dimensions for the latent space results in generating better grasps. Qualitative results are provided in Fig S.6.

For RefineNet we take the output of the CoarseNet (MANO parameters) and first compute the distances of MANO vertices to the object vertices. We then pass the distances with the MANO parameters to the network. RefineNet refines the input grasp through 3 iterations. The CoarseNet and RefineNet are trained for 16 and 23 epochs respectively with the learning rate starting from 5e - 4, decreasing on validation error plateau to 0.1 times, and early stopping after 8 epochs with no improvement in validation error. Both networks are trained separately.

S.9.4 Filtering out Unreliable Turkers

As mentioned in the main paper (Sec. 4), along with ground-truth (GRAB) and GrabNet-generated grasps, we pass to Turkers noisy grasps generated by perturbing ground-truth ones. These noisy grasps are our test for spotting unreliable Turkers, that either select their answers randomly or misunderstand the task. Specifically, we remove the ones that gave a rating of 3 or more (indicating good realism) for at least 20% of these noisy grasps. In total we removed 54 out of 170 Turkers.

S.9.5 Heatmaps for Various Intents and Fine-Grained Numbers

Similar to Sec 3.5 of the main manuscript, here Fig. S.9 provides additional fine-grained numbers for in-contact parts of the body. Each row corresponds to an intent in the GRAB dataset. For each intent, the right column shows the contact percentage and "heatmap" for the right hand, left hand, and head across all frames and relative to *all* body vertices. In the three left columns, the "heatmaps" and percentages are relative to only *each part's* vertices and for only the frames for which these parts are in contact (left hand, right hand, and head), for visualization purposes. For example, for the "use" sequences (second row in Fig S.9), the right hand was in contact for 90.62% of all frames, and in those contact frames the thumb fingertip was in contact for 99.88% of them.

S.10 Bias from MoCap Markers

A natural question arises - are subjects biased in their grasps by MoCap markers? We empirically place more markers in areas less likely to be contacted, according to object affordances. To account for potential occlusions, though, we have to place some markers in other areas as well. For this reason, we still expect our markers to be contacted.

10 O. Taheri et al.

Our subjects did not complain about discomfort or bias, yet we need more evidence for this. Apart from the analysis in the main manuscript (see Sec 3.5 and Fig. 5), here we perform k-means clustering (k=20) on our grasps, and visualize each cluster center, i.e. a grasping hand, and the grasped object. We observe that several clusters (typically 3-6 out of 20) show that fingers do come in contact with markers. Figure S.10 shows for 5 objects (rows) 3 representative contacting clusters (columns). We believe this is good additional empirical evidence that our 1.5 mm radius hemi-spherical markers cause no or minimal bias.

S.11 Influence of Contact Heuristic Thresholds

We use several heuristics to determine contact frames, see Sec. 3.3 in the main manuscript. For the contact "heatmap" analysis we take all the contact frames for which object is being manipulated, i.e. it is off the table. Because the heatmap is integrated over many frames, small variations in the heuristics have little impact on the contact patterns.

To show this empirically, we perform a sensitivity analysis by changing our thresholds. Figure S.11 shows "heatmaps" for "use" sequences for several setups, following the format of Fig. 5 of the main manuscript. The results verify our hypothesis that the heuristics have minimal influence.

References

- Bogo, F., Romero, J., Pons-Moll, G., Black, M.J.: Dynamic faust: Registering human bodies in motion. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
- Brahmbhatt, S., Ham, C., Kemp, C.C., Hays, J.: ContactDB: Analyzing and predicting grasp contact via thermal imaging. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
- Cudeiro, D., Bolkart, T., Laidlaw, C., Ranjan, A., Black, M.: Capture, learning, and synthesis of 3D speaking styles. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
- 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: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
- Hampali, S., Oberweger, M., Rad, M., Lepetit, V.: HO-3D: A multi-user, multiobject dataset for joint 3D hand-object pose estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
- Hasson, Y., Varol, G., Tzionas, D., Kalevatykh, I., Black, M.J., Laptev, I., Schmid, C.: Learning joint reconstruction of hands and manipulated objects. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
- Li, T., Bolkart, T., Black, M.J., Li, H., Romero, J.: Learning a model of facial shape and expression from 4D scans. ACM Transactions on Graphics (TOG) 36(6), 194:1–194:17 (2017)

- Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: A skinned multi-person linear model. ACM Transactions on Graphics (TOG) 34(6), 248:1–248:16 (2015)
- Mahmood, N., Ghorbani, N., F. Troje, N., Pons-Moll, G., Black, M.J.: AMASS: Archive of motion capture as surface shapes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
- Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3D hands, face, and body from a single image. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
- Prokudin, S., Lassner, C., Romero, J.: Efficient learning on point clouds with basis point sets. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
- Ranjan, A., Bolkart, T., Sanyal, S., Black, M.J.: Generating 3D faces using convolutional mesh autoencoders. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018)
- Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. ACM Transactions on Graphics (TOG) 36(6), 245:1– 245:17 (2017)
- Zhou, Y., Barnes, C., Lu, J., Yang, J., Li, H.: On the continuity of rotation representations in neural networks. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)



Fig. S.5: GrabNet Architecture. For the encoder input, we concatenate the BPS representation of the object with MANO parameters, while for decoder input we concatenate it with a sample from latent space. The decoder gives the MANO hand parameters which we pass to the MANO model to obtain the 3D hand mesh.



Fig. S.6: Visualization of 10 different grasps (rows) generated by GrabNet for 6 unseen objects (columns). Conditioned on the BPS_o representation of unseen 3D object shapes, we sample from the learned 16 dimensional latent grasping space Z of CoarseNet (see Fig. 8 of the paper). We then concatenate BPS_o to the Z sample, and pass them to the decoder of CoarseNet, that outputs the coarse grasping MANO hand model parameters. We then pass the coarse grasps to the RefineNet to get the final grasps. Some failure cases are highlighted with red. Different viewpoint for the results of Fig. S.7, S.8.



Fig. S.7: Visualization of 10 different grasps (rows) generated by GrabNet for 6 unseen objects (columns). Conditioned on the BPS_o representation of unseen 3D object shapes, we sample from the learned 16 dimensional latent grasping space Z of CoarseNet (see Fig. 8 of the paper). We then concatenate BPS_o to the Z sample, and pass them to the decoder of CoarseNet, that outputs the coarse grasping MANO hand model parameters. We then pass the coarse grasps to the RefineNet to get the final grasps. Some failure cases are highlighted with red. Different viewpoint for the results of Fig. S.6, S.8.



Fig. S.8: Visualization of 10 different grasps (rows) generated by GrabNet for 6 unseen objects (columns). Conditioned on the BPS_o representation of unseen 3D object shapes, we sample from the learned 16 dimensional latent grasping space Z of CoarseNet (see Fig. 8 of the paper). We then concatenate BPS_o to the Z sample, and pass them to the decoder of CoarseNet, that outputs the coarse grasping MANO hand model parameters. We then pass the coarse grasps to the RefineNet to get the final grasps. Some failure cases are highlighted with red. Different viewpoint for the results of Fig. S.6, S.7.



Fig. S.9: Contact "heatmaps" and percentages for all intents in GRAB, for various body parts. Each row corresponds to an intent in the GRAB dataset. For each intent, the right column shows the results for each part (right hand, left hand, and head) across all frames and relative to *all* body vertices. In the three left columns, the results are relative to only *each part's* vertices and for only frames for which these parts are in contact (left hand, right hand, and head), for visualization purposes.



Fig. S.10: Do subjects avoid markers? To answer this, we perform k-means clustering (k=20) on our grasps, and visualize each cluster center, i.e. a grasping MANO (green), and the grasped object (blue). We observe that several cluster centers (columns) per object (rows) show that subjects contact MoCap markers (red); here we show 3 clusters for 5 objects. For fingers that contact markers we render only the vertices, to allow to see the markers (best viewed on screen).



Fig. S.11: Sensitivity analysis for contact heuristics. We follow the format of Fig. 5 of the main manuscript, and show "heatmaps" and contact likelihoods in percentages % for a subset of "all" sequences with different setups (columns), as indicated in the labels. The symbol H denotes the minimum difference between the object's vertical position from its initial one (resting on a table). The symbol C denotes the minimum number of object vertices that we require to be in contact with each finger. The figure shows that threshold choices have minimal effect when integrated over many frames to create "heatmaps".