Reconstructing Hand-Held Objects in 3D from Images and Videos

Author Names Omitted for Anonymous Review. Paper-ID [1]

***Abstract*—Objects manipulated by the hand (i.e., manipulanda) are particularly challenging to reconstruct from Internet videos. Not only does the hand occlude much of the object, but also the object is often only visible in a small number of image pixels. At the same time, two strong anchors emerge in this setting: (1) estimated 3D hands help disambiguate the location and scale of the object, and (2) the set of manipulanda is small relative to all possible objects. With these insights in mind, we present a scalable paradigm for hand-held object reconstruction that builds on recent breakthroughs in large language/vision models and 3D object datasets. Given a monocular RGB video, we aim to reconstruct hand-held object geometry in 3D, over time. In order to obtain the best performing single frame model, we first present MCC-Hand-Object (MCC-HO), which jointly reconstructs hand and object geometry given a single RGB image and inferred 3D hand as inputs. Subsequently, we prompt a text-to-3D generative model using a VLM to retrieve a 3D object model that matches the object in the image(s); we call this alignment Retrieval-Augmented Reconstruction (RAR). RAR provides unified object geometry across all frames, and the result is rigidly aligned with both the input images and 3D MCC-HO observations in a temporally consistent manner. Experiments demonstrate that our approach achieves state-of-the-art performance on lab and Internet image/video datasets.**

I. Introduction

Recovering 3D hand-object interactions from visual data is an important problem in both computer vision and robotics. In this paper, we present a novel approach to recovering hand-object interactions in 3D from monocular RGB images and videos, focusing in particular on reconstructing hand-held object geometry. See Figure 1. Given an RGB image and estimated 3D hand, we train a transformer-based model, MCC-Hand-Object (MCC-HO), that jointly infers 3D hand and object geometry. This geometry is represented as a neural implicit surface composed of occupancy, color, and hand-object segmentation. By adapting our model architecture from an existing object reconstruction model, MCC [1], we make use of a learned object prior by fine-tuning an MCC model that is pretrained on the CO3Dv2 dataset [2]. Experiments show that our model outperforms existing methods for hand-held object reconstruction on available datasets containing 3D labels, including DexYCB, MOW, and HOI4D [3]–[5] (Table I, Figure 3).

Given the limited size of existing 3D hand-object datasets, we additionally leverage vision-language models [6], [7] and 3D generative models [8]–[11] in order to improve upon a domain-specific approach. In conjunction with MCC-HO, we propose Retrieval-Augmented Reconstruction (RAR), a method for automatically retrieving object models using large language/vision models to improve network-inferred geometry.

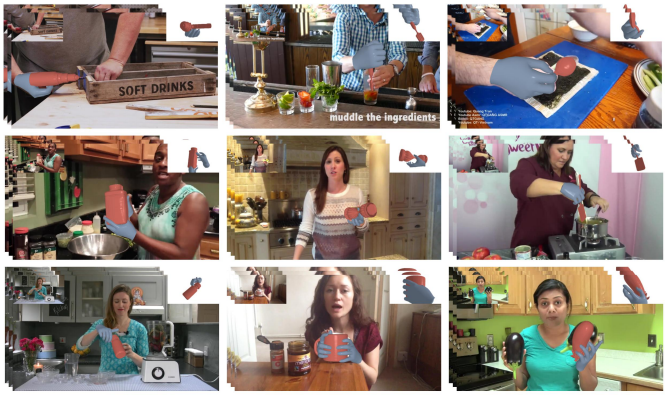

Fig. 1: We present a scalable approach to hand-held object reconstruction from RGB images or videos that is guided by object recognition and retrieval. Results demonstrate that our method is able to generate realistic object geometry that is also faithful to visual observations and consistent across frames.

Specifically, we prompt a VLM [6] to recognize the hand-held object in the image and provide a detailed text description. This description is passed to a text-to-3D generative model [8] to obtain a 3D object, which is then rigidly aligned with the input image or video using a combination of DINOv2 [12] and inferred 3D cues from MCC-HO. We demonstrate that RAR leads to quantitative improvements in 3D reconstruction (Table II), as well as more realistic object geometries compared to existing image-to-3D methods (Figures 3 and 4).

II. Method Overview

Our approach to reconstructing hand-object interactions from monocular RGB video combines feedforward network inference (MCC-HO) with object model retrieval (RAR) and subsequent rigid alignment. See Figure 2. First, our transformer-based model, MCC-HO, estimates 3D hand-object geometry from single images (Section III). Simultaneously, a 3D object model is automatically synthesized by a text-to-3D generative model [8] through a technique we coin Retrieval-Augmented Reconstruction (Section IV). Finally, the 3D object model is rigidly aligned in a temporally consistent manner (Section V).

III. Estimating Hand-Object Geometry

In the first stage, hand and object geometry are jointly inferred by training a model adapted from MCC [1] for the specific task of hand-object reconstruction. Our model, MCC-Hand-Object (MCC-HO), has to deal with two major

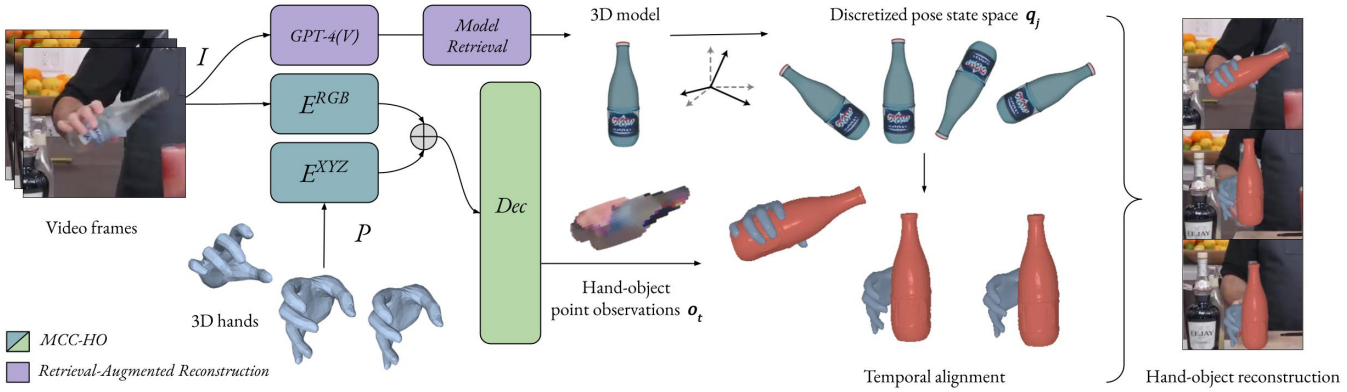

Fig. 2: Given an RGB video and estimated 3D hands, our method reconstructs 3D hand-held object trajectories. First, MCC-HO is used to predict hand and object point clouds for each frame (Section III). Then, a single 3D model for the object is obtained using Retrieval-Augmented Reconstruction (Section IV). The 3D object model is rigidly aligned with DINOv2 [12] image features and network-inferred geometry in a temporally consistent manner via our Viterbi algorithm (Section V).

challenges. First, MCC assumes RGB-D images as input, whereas our model needs to work with RGB input, since our ultimate goal is to reconstruct from Internet videos. Second, we have to find a way to cope with the significant domain shift between the training datasets. MCC was trained on the CO3D dataset [13], which consists of clean, 360-degree videos where each object is unoccluded and placed on a solid surface. For MCC-HO, our training data is a relatively small collection of in-the-wild and lab datasets of hand-object interactions, where the hand-held object is at least partially occluded in the vast majority of images.

The inputs to MCC-HO are a single RGB image and 3D hand geometry. Hand and object segmentation masks can be obtained either from ground truth labels or Segment Anything (SAM/SAM2) [14], [15], and the combined hand-object mask determines the bounding box that is used to crop/resize the input image. Hand geometry is parameterized by the MANO hand model [16], which is composed of pose parameters $\theta \in \mathbb{R}^{48}$ and shape parameters $\beta \in \mathbb{R}^{10}$. The MANO hand skeleton contains 21 joints. At training time, the ground truth 3D hand is used. At test time, HaMeR [17] is used to infer the hand geometry.

The inferred geometry is represented as a neural implicit function $\rho(x)$ modeling the hand and object jointly, such that for any 3D location x, ρ returns the occupancy probability $\sigma(x)$, RGB color $c(x)$, and segmentation label $m(x)$:

$$\rho(x) = (\sigma(x), c(x), m(x)) \tag{1}$$

where $\sigma(x) \in [0, 1]$, $c(x) \in [0, 1]^3$, and $m(x) \in \{0, 1, 2\}$ to indicating a background, hand, or object point. The scale of the object geometry is normalized with respect to the input 3D hand.

A. Architecture

MCC-HO has an encoder-decoder architecture that is composed of transformers [18] and outputs an implicit representation of hand-object geometry. The input images and 3D hand are passed to the encoder to compute a feature map that is a function of both image and hand embeddings. These features are then used to condition the decoder, which maps any 3D location to occupancy, color, and hand-object segmentation. During training, a set of query 3D points Q are uniformly sampled within the bounds of a normalized volume. At test time, vertices of a voxel grid are passed as query points.

Encoder. The encoder takes as input a single RGB image $I \in \mathbb{R}^{H \times W \times 3}$ and a posed MANO hand mesh defined in camera world space with vertices $v_h \in \mathbb{R}^{778 \times 3}$ and faces $f_h \in \mathbb{R}^{1538 \times 3}$. The camera intrinsics are either known or consistent with the estimated 3D hand (*e.g.*, the camera parameters used during HaMeR inference [17]). In order to jointly encode the image and hand geometry, the hand mesh is rasterized and per-pixel 3D points for the hand are sampled.

The image I and hand points P are encoded into a single representation R, where:

$$R \coloneqq f(E^{RGB}(I), E^{XYZ}(P)) \in \mathbb{R}^{N^{enc} \times C}, \tag{2}$$

and E^{RGB} and E^{XYZ} are identical to the transformers proposed in MCC [1], though we define P differently, *i.e.*, without an input depth image. The image transformer E^{RGB} has a Vision Transformer (ViT) architecture [19] with 16×16 patch embeddings. The point transformer E^{XYZ} uses a self-attention-based patch embedding design that differentiates between seen and unseen pixel points.

Decoder. The decoder takes as input the encoder output, R, as well as a set of query points $Q \in \mathbb{R}^{N_q \times 3}$ (we choose $N_q = 1024$). These inputs are passed to a transformer with architecture adopted from MCC and inspired by Masked Autoencoders (MAE) [20]. Each query token is passed to three output heads that infer binary occupancy σ, RGB color c, and hand-object segmentation m, such that:

$$Decoder(R, Q) \coloneqq (\sigma(Q), c(Q), m(Q)). \tag{3}$$

The decoder layers that output occupancy and color align with MCC, but MCC-HO has an additional linear layer that infers

multiclass segmentation of each point into either background, hand, or object labels.

Losses. The training loss is a combination of occupancy, color, and segmentation:

$$\mathcal{L} = \mathcal{L}_\sigma + \mathcal{L}_c(\sigma_{gt}) + \mathcal{L}_m \tag{4}$$

The occupancy loss $\mathcal{L}_\sigma$ is a binary cross-entropy loss comparing the predicted and ground truth occupancies of each query point. The color loss $\mathcal{L}_c$ is a 256-way classification loss computed for query points that have ground truth occupancy of 1. The segmentation loss $\mathcal{L}_m$ is 3-way classification loss for each query point corresponding to background (empty space), hand, and object labels.

IV. Retrieval-Augmented Reconstruction

In order to further improve upon network-inferred object geometry, we exploit existing foundation models by using a VLM [6] and a text-to-3D generative model [8] to automatically detect and "retrieve" a 3D object corresponding to the hand-held object in an image. We call this technique Retrieval-Augmented Reconstruction (RAR), somewhat in analogy to Retrieval-Augmented Generation in natural language processing [21].

A. Rigid Alignment

After a 3D object model is retrieved, it is rigidly aligned with the input visual data and MCC-HO predictions. We begin by considering the single image case. First, the object scale is estimated using the predicted MCC-HO object point cloud, which is a reasonable approximation because the prediction is grounded by the input 3D hand. If the MCC-HO point cloud is $X \in \mathbb{R}^{N\times 3}$ and a sampled point cloud of the retrieved object model is $Y \in \mathbb{R}^{N\times 3}$, then the scale is defined as $s = \sqrt{\lambda_X^0}/\sqrt{\lambda_Y^0}$ where λ_X^0 is the largest eigenvalue of $X^T X$ and λ_Y^0 is similarly defined for Y.

The remaining parameters to be optimized are object rotation R and translation T. In order to (1) guarantee a globally optimal solution while (2) maintaining fixed compute time, we discretize the space of all possible parameter values to construct finite state spaces $\{\mathcal{R}, \mathcal{T}\}$. The 3D rotation group, SO(3), is discretized using the unit cube subdivision strategy proposed in [22]. Translation is defined as an offset from the mean location of the MCC-HO point cloud $\mu(X)$, and the space is discretized by constructing a small voxel grid centered at $\mu(X)$. In order to avoid intractably large state space sizes (*e.g.*, $\mathbb{R}^3 \times \mathbb{R}^3$), we first optimize for rotation and then translation. For each parameter, we select the state that results in the lowest error using the sum of the following two metrics.

MCC-HO Chamfer Distance. The Chamfer Distance (CD) between the MCC-HO point cloud and a transformed object model is the usual 3D evaluation metric. Note that when optimizing for rotation, the object model (centered at the origin) is first rotated and then translated to the mean location of the MCC-HO point cloud.

DINOv2 PCA Similarity. The MCC-HO point cloud gives us a reasonable estimate for object pose, but using this signal alone may not result in poses that align well with the input image (particularly for geometry with symmetries and in-the-wild objects). Thus, we additionally use DINOv2 [12] features to determine visual similarities between a transformed object model and the input image. For each rotation/translation state, the transformed object mesh is rendered using Pytorch3D [23]. A PCA basis is constructed using the DINOv2 features of the input image, and the first three PCA components are computed for both the rendered features $\mathcal{F}_j$ and input image features $\mathcal{F}_0$. We compute the cosine similarity between the two as

$$E_{DINO} = 1 - \frac{1}{2}\left(\frac{\text{PCA}(\mathcal{F}_j)\cdot\text{PCA}(\mathcal{F}_0)}{\max(||\text{PCA}(\mathcal{F}_j)||_2 \cdot ||\text{PCA}(\mathcal{F}_0)||_2), \epsilon)} + 1\right) \tag{5}$$

where $\mathcal{F}_j$ and $\mathcal{F}_0$ are both masked by the input image object silhouette. When ground truth object masks are not available, we use SAM 2 [15].

V. From Images to Videos

If the input is a video stream, we can further exploit the additional cue of temporal smoothness—we expect adjacent frames to have rotation and translation states that are close to each other. Given a sequence of video frames with timesteps t, we jointly optimize for per-frame rotations $R(t)$ and subsequently per-frame translations $T(t)$. Aiming for robustness and efficiency, we propose using the Viterbi algorithm [24]–[26] to solve for a global solution to rigid object alignment across all frames of a video sequence. The Viterbi algorithm is a dynamic programming approach to obtaining the maximum *a posteriori* probability of the most likely sequence of hidden states $Q = q_1, q_2, \ldots, q_N$ that results in a sequence of observed events $O = o_1, o_2, \ldots, o_N$. The hidden states q_t can take on values from a finite set of parameters $\mathcal{Q}$ (where in our case, $\mathcal{Q} = \mathcal{R}$ or $\mathcal{Q} = \mathcal{T}$). Following the single image case, the observations o_t are a combination of the MCC-HO inferred object point clouds and the DINOv2 image features.

Each cell of the Viterbi trellis, $v_t(j)$, is the probability of ending up in state j after seeing the first t observations and passing through the most probable state sequence $q_1, \ldots, q_{t-1}$. Given the probability of being in every state at time $t-1$, the Viterbi probabilities are computed via:

$$v_t(j) = \max_{i=1}^{|\mathcal{Q}|} v_{t-1}(i) a_{ij} b_j(o_t), \tag{6}$$

where a_{ij} is the transition probability from the previous state q_i to the current state q_j, and $b_j(o_t)$ is the state observation likelihood given the current state j. In particular, a_{ij} is Rodrigues' rotation error when optimizing for rotations and Euclidean distance when optimizing for translations. The emission cost $b_j(o_t)$ is the sum of the two metrics defined in the single image discussion, *e.g.*, CD with respect to the MCC-HO point clouds and DINOv2 PCA similarity with respect to the input images.

VI. Experiments

A. Quantitative Evaluation

We quantitatively evaluate MCC-HO on a variety of labeled hand-object datasets and in comparison to existing model-free

	DexYCB				MOW				HOI4D			
	F-5 (↑)	F-10 (↑)	CD (↓)	Vol (↓)	F-5 (↑)	F-10 (↑)	CD (↓)	Vol (↓)	F-5 (↑)	F-10 (↑)	CD (↓)	Vol (↓)
HO [27]	0.24	0.48	4.76	11.8	0.03	0.06	49.8	78.2	0.28	0.51	3.86	12.7
IHOI [28]	-	-	-	-	0.13	0.24	23.1	20.4	0.42	0.70	2.7	**0.91**
HORSE [29]	0.23	0.42	6.97	3.63	0.11	0.23	24.5	17.0	0.26	0.45	6.69	1.39
MCC-HO (Ours)	**0.36**	**0.60**	**3.74**	**2.42**	**0.15**	**0.31**	**15.2**	**13.8**	**0.52**	**0.78**	**1.36**	1.94

TABLE I: We compare our method, MCC-HO, to prior works on held-out test images from DexYCB, MOW, and HOI4D. Chamfer Distance (cm^2), F-score (5mm, 10mm), and Intersection Volume using 0.5cm voxels (as in HO [27], IHOI [28]) are reported.

Method	F-5 (↑)	F-10 (↑)	CD (↓)
G-HOP [30]	0.61	0.89	0.8
MCC-HO + RAR	**0.74**	**0.91**	**0.64**

TABLE II: Quantitative comparison to a state-of-the-art video-based hand-held object reconstruction method using the HOI4D test dataset. Mean metrics are reported.

reconstruction approaches [27]–[29]. See Table I. In order to sample points more densely in hand-object regions, the dimensions and granularity of the voxel grid used to query MCC-HO are determined from an initial coarse prediction using a default voxel grid. For DexYCB and HOI4D, we use every 5th frame of each test video sequence. The results for IHOI [28] and HO [27] evaluated on the MOW dataset are obtained from the IHOI and DiffHOI [31] papers; otherwise, we used available public repositories of prior work. Our method achieves state-of-the-art performance for the Chamfer Distance and F-score metrics on all three datasets, implying that our technique is not overfitted to any particular dataset. We conjecture that increasing the number of unique object instances seen during training is important to model generalization. Thus, the pretrained MCC model provides a strong object prior for initialization.

We also compare to G-HOP [30], a state-of-the-art video-based approach for RGB hand-held object reconstruction in Table II. Our method (using generated Genie objects for RAR) outperforms the generalist G-HOP model (that was trained on seven hand-object datasets [3], [5], [32]–[35]) in all three metrics.

B. Qualitative Evaluation

We qualitatively evaluate (1) the reconstruction quality of MCC-HO and (2) how well retrieved object models using RAR can be rigidly aligned to the predicted point clouds in 3D (a necessary and sufficient condition for downstream tasks). Figure 3 shows qualitative comparisons to existing work, where the combination of MCC-HO + RAR allows us to obtain more realistic object geometry.

1) Retrieval-Augmented Reconstruction: The ultimate goal of RAR is to provide a scalable paradigm for creating large hand-object 3D datasets; to this end, we explore the potential to scale up our approach on unlabeled images and videos. Figure 4 shows our results for 9 images from 100DOH [36] that do not have 3D ground truth (*e.g.*, not in MOW). Estimated 3D hands for each image are obtained using HaMeR [17]. At test time, our method does not require an input hand-object segmentation mask [4], [37], depth image [1], or hand joint locations [28], [38].

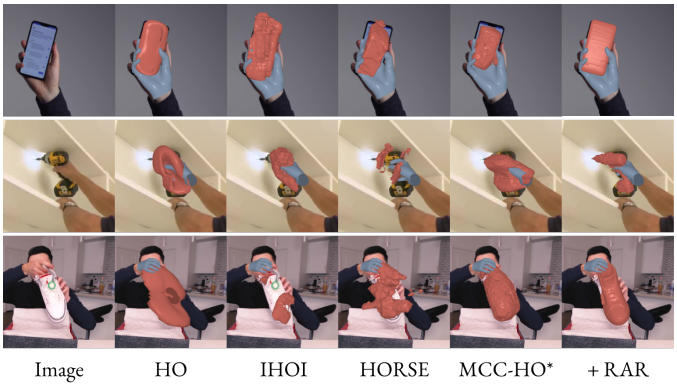

Fig. 3: Qualitative comparisons on MOW. *Note that the MCC-HO point clouds are rendered as a mesh via Poisson surface reconstruction, which can introduce artifacts not attributed to our method. The last column is MCC-HO + RAR.

Fig. 4: Our approach applied to in-the-wild images from the 100 Days of Hands dataset [36]. For each image, we estimate 3D hands using HaMeR [17], inference object geometry using MCC-HO and retrieve a 3D model using RAR.

VII. Conclusion

In this paper, we present a novel paradigm for reconstructing hand-held objects that combines the respective benefits of model-free and model-based prediction. Our transformer-based model, MCC-HO, is trained to predict hand-object geometry given a single RGB image and estimated 3D hand. Experiments conducted using the DexYCB, MOW, and HOI4D datasets all demonstrate that MCC-HO achieves state-of-the-art performance in hand-held object reconstruction. We additionally present Retrieval-Augmented Reconstruction (RAR), an automatic method for object model retrieval that leverages recent advances in large language/vision models and 3D object datasets. These two approaches can be combined to scale the amount of labeled hand-object interaction data, as suggested by our results using unlabeled 100DOH videos [36].

REFERENCES

[1] C.-Y. Wu, J. Johnson, J. Malik, C. Feichtenhofer, and G. Gkioxari, "Multiview compressive coding for 3d reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9065–9075.

[2] J. Reizenstein, R. Shapovalov, P. Henzler, L. Sbordone, P. Labatut, and D. Novotny, "Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 901–10 911.

[3] Y.-W. Chao, W. Yang, Y. Xiang, P. Molchanov, A. Handa, J. Tremblay, Y. S. Narang, K. Van Wyk, U. Iqbal, S. Birchfield *et al.*, "Dexycb: A benchmark for capturing hand grasping of objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9044–9053.

[4] Z. Cao, I. Radosavovic, A. Kanazawa, and J. Malik, "Reconstructing hand-object interactions in the wild," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 12 417–12 426.

[5] Y. Liu, Y. Liu, C. Jiang, K. Lyu, W. Wan, H. Shen, B. Liang, Z. Fu, H. Wang, and L. Yi, "Hoi4d: A 4d egocentric dataset for category-level human-object interaction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 21 013–21 022.

[6] "Openai. gpt-4v(ision) system card." 2023.

[7] G. Comanici, E. Bieber, M. Schaekermann, I. Pasupat, N. Sachdeva, I. Dhillon, M. Blistein, O. Ram, D. Zhang, E. Rosen *et al.*, "Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities," *arXiv preprint arXiv:2507.06261*, 2025.

[8] "Luma ai. capture 3d. https://lumalabs.ai." 2023.

[9] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall, "Dreamfusion: Text-to-3d using 2d diffusion," *arXiv preprint arXiv:2209.14988*, 2022.

[10] Z. Chen, F. Wang, Y. Wang, and H. Liu, "Text-to-3d using gaussian splatting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 401–21 412.

[11] X. Long, Y.-C. Guo, C. Lin, Y. Liu, Z. Dou, L. Liu, Y. Ma, S.-H. Zhang, M. Habermann, C. Theobalt *et al.*, "Wonder3d: Single image to 3d using cross-domain diffusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 9970–9980.

[12] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

[13] J. Reizenstein, R. Shapovalov, P. Henzler, L. Sbordone, P. Labatut, and D. Novotny, "Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction," in *International Conference on Computer Vision*, 2021.

[14] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," *arXiv preprint arXiv:2304.02643*, 2023.

[15] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson *et al.*, "Sam 2: Segment anything in images and videos," *arXiv preprint arXiv:2408.00714*, 2024.

[16] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: Modeling and capturing hands and bodies together," *arXiv preprint arXiv:2201.02610*, 2022.

[17] G. Pavlakos, D. Shan, I. Radosavovic, A. Kanazawa, D. Fouhey, and J. Malik, "Reconstructing hands in 3d with transformers," *arXiv preprint arXiv:2312.05251*, 2023.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[20] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 16 000–16 009.

[21] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.

[22] G. Kurz, F. Pfaff, and U. D. Hanebeck, "Discretization of so (3) using recursive tesseract subdivision," in *2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. IEEE, 2017, pp. 49–55.

[23] N. Ravi, J. Reizenstein, D. Novotny, T. Gordon, W.-Y. Lo, J. Johnson, and G. Gkioxari, "Accelerating 3d deep learning with pytorch3d," *arXiv preprint arXiv:2007.08501*, 2020.

[24] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967.

[25] G. D. Forney, "The viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.

[26] D. Jurafsky, *Speech & language processing*. Pearson Education India, 2000.

[27] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev, and C. Schmid, "Learning joint reconstruction of hands and manipulated objects," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 11 807–11 816.

[28] Y. Ye, A. Gupta, and S. Tulsiani, "What's in your hands? 3d reconstruction of generic objects in hands," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 3895–3905.

[29] A. Prakash, M. Chang, M. Jin, R. Tu, and S. Gupta, "3d reconstruction of objects in hands without real world 3d supervision," in *European Conference on Computer Vision (ECCV)*, 2024.

[30] Y. Ye, A. Gupta, K. Kitani, and S. Tulsiani, "G-hop: Generative hand-object prior for interaction reconstruction and grasp synthesis," *arXiv preprint arXiv:2404.12383*, 2024.

[31] Y. Ye, P. Hebbar, A. Gupta, and S. Tulsiani, "Diffusion-guided reconstruction of everyday hand-object interaction clips," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 19 717–19 728.

[32] S. Brahmbhatt, C. Tang, C. D. Twigg, C. C. Kemp, and J. Hays, "Contactpose: A dataset of grasps with object contact and hand pose," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*. Springer, 2020, pp. 361–378.

[33] E. Corona, A. Pumarola, G. Alenya, F. Moreno-Noguer, and G. Rogez, "Ganhand: Predicting human grasp affordances in multi-object scenes," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 5031–5041.

[34] O. Taheri, N. Ghorbani, M. J. Black, and D. Tzionas, "Grab: A dataset of whole-body human grasping of objects," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*. Springer, 2020, pp. 581–600.

[35] L. Yang, K. Li, X. Zhan, F. Wu, A. Xu, L. Liu, and C. Lu, "Oakink: A large-scale knowledge repository for understanding hand-object interaction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 20 953–20 962.

[36] D. Shan, J. Geng, M. Shu, and D. F. Fouhey, "Understanding human hands in contact at internet scale," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9869–9878.

[37] A. Patel, A. Wang, I. Radosavovic, and J. Malik, "Learning to imitate object interactions from internet videos," *arXiv preprint arXiv:2211.13225*, 2022.

[38] H. Choi, N. Chavan-Dafle, J. Yuan, V. Isler, and H. Park, "Handnerf: Learning to reconstruct hand-object interaction scene from a single rgb image," *arXiv preprint arXiv:2309.07891*, 2023.