arXiv:2503.19307v1 [cs.CV] 25 Mar 2025

Analyzing the Synthetic-to-Real Domain Gap in 3D Hand Pose Estimation

Zhuoran Zhao[1*] Linlin Yang[2†] Pengzhan Sun[3] Pan Hui[1] Angela Yao[3]
[1]The Hong Kong University of Science and Technology (Guangzhou), China
[2]Communication University of China, China [3]National University of Singapore, Singapore
zzhao074@connect.hkust-gz.edu.cn, lyang@cuc.edu.cn, panhui@ust.hk,
{pengzhan, ayao}@comp.nus.edu.sg

Abstract

Recent synthetic 3D human datasets for the face, body, and hands have pushed the limits on photorealism. Face recognition and body pose estimation have achieved state-of-the-art performance using synthetic training data alone, but for the hand, there is still a large synthetic-to-real gap. This paper presents the first systematic study of the synthetic-to-real gap of 3D hand pose estimation. We analyze the gap and identify key components such as the forearm, image frequency statistics, hand pose, and object occlusions. To facilitate our analysis, we propose a data synthesis pipeline to synthesize high-quality data. We demonstrate that synthetic hand data can achieve the same level of accuracy as real data when integrating our identified components, paving the path to use synthetic data alone for hand pose estimation. Code and data are available at: https://github.com/delaprada/HandSynthesis.git.

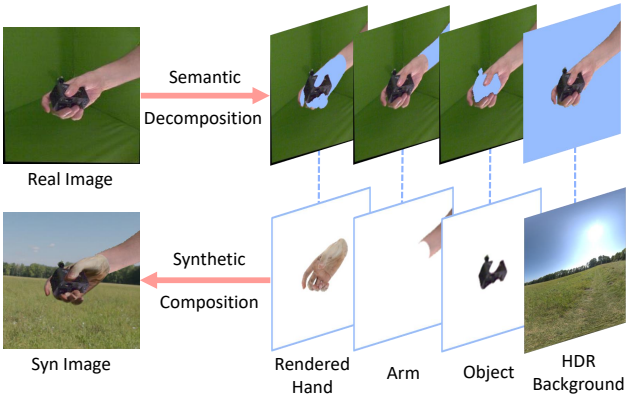

Figure 1. We present a systematic study of the synthetic-to-real gap for 3D hand pose estimation by decomposing the synthetic data to establish associations between hand image components and predictions.

1. Introduction

3D hand pose estimation from monocular RGB images plays an important role in computer vision, human-computer interaction, and AR/VR applications [5, 16, 18]. It is challenging since the hand has severe occlusion problems and depth ambiguity. Most recent methods [11, 25, 32] rely heavily on 3D ground-truth labels for training, but obtaining these labels is expensive and time-consuming. Therefore, there is strong interest in using synthetic data, as it does not require any annotation effort. In face recognition [2, 45] and 3D human pose estimation [3], state-of-the-art is achievable with only synthetic training sets. However, there is still a large synthetic-to-real gap for 3D hand pose estimation [14, 17, 22, 50], despite recent efforts to diversify backgrounds, pose distributions, camera viewpoints, and hand textures for synthetic data. Pre-training on synthetic and fine-tuning with real data, or mixed synthetic and real training, is still much better than training with only synthetic data [10, 14, 17, 22, 34]. These observations beg the question: What contributes to the synthetic-to-real gap in 3D hand pose estimation?

Many synthetic hand datasets exist but are difficult to analyze since they differ from real hand datasets across multiple aspects (*e.g.*, pose distribution, background, texture, *etc.*), which cannot be separated. Therefore, we propose a data synthesis pipeline to facilitate our study. The pipeline efficiently synthesizes high-quality data with diverse backgrounds, lighting, and hand textures. With full control over the rendering pipeline, we can decompose synthetic data to analyze associations between different hand image components and predictions to enhance synthetic-to-real gap analysis (see Fig. 1). Moreover, since our pipeline can synthesize controllable pair-wise data, it can benefit future research for further analysis and enhance model performance for downstream applications.

We analyze the synthetic-to-real gap from four aspects:

*Done while the first author was a research intern at NUS@CVML
†Corresponding author

Table 1. Representative synthetic hand datasets. HDR denotes high dynamic range scenes. HDR Res. denotes HDR background resolution, with higher values being more realistic and less blurry. For lighting, "manual" denotes manually defined settings while "dynamic" denotes diverse lighting conditions consistently applied across the foreground hand and background scene. "NS" means normal and specular maps for hand texturing. "GR" means the hand data generation and image rendering pipeline.

Dataset	Data Size	Background	HDR Type	HDR Res.	Lighting	Texture	NS	Template	Hand Type	GR
RHD [55]	44K	static	-	-	manual	limited	✗	Mixamo	single hand	✗
ObMan [17]	154K	static	-	-	manual	limited	✗	MANO	hand object	✓
DARTset [14]	800K	static	-	-	manual	diverse	✗	DART	single hand	✓
Ego3D [24]	55K	static	-	-	manual	limited	✗	Mixamo	two hands	✗
Re:InterHand [34]	1.5M	HDR	Indoor	7K	dynamic	diverse	✗	MANO	two hands	✗
RenderIH [22]	1M	HDR	Indoor & Outdoor	1K	dynamic	diverse	✗	A-MANO	two hands	✓
Ours	460K	HDR	Indoor & Outdoor	4K	dynamic	diverse	✓	NIMBLE	single hand	✓

appearance, hand pose, object occlusion, and hand skeleton topology, which are essential for hand pose estimation. From the appearance perspective, we first find that the presence of the forearm is important for pose estimation models to locate the wrist, affecting overall pose accuracy. Secondly, amplitude spectrum augmentation [9] is crucial for enhancing the model's robustness when the domain shifts from synthetic to real. Thirdly, increasing the background and hand texture diversity is important in reducing the synthetic-to-real gap, but their influence saturates beyond a certain amount. From the pose perspective, we observe that the pose difference between synthetic and real data affects the synthetic-to-real performance. However, performance gains tend to plateau after a certain threshold as more real hand poses are used for data synthesis. Notably, we find that using only a subset of real hand poses for synthesizing hand data enables the model trained on such data to achieve comparable performance to using all real hand poses. This outcome demonstrates that the data synthesized by our pipeline effectively facilitates learning hand representations.

Unlike the human face and body, the hand has greater self-occlusion and interacting object occlusions. Yet, only a few synthetic hand datasets feature hand-object interactions. Incorporating object occlusions into synthetic training data improves model generalization on real-world test samples. Additionally, we observe that pose estimation models can associate hand poses with specific objects. Modeling specific object occlusions as a prior can improve performance when encountering similar objects. Finally, synthetic and real datasets often use different hand skeleton topologies [13, 35]. This discrepancy can cause inaccurate evaluation and label adaptation is necessary.

By incorporating different hand image components, models trained with only our synthesized data can achieve state-of-the-art performance compared with real data. Moreover, models trained on a mixture of real and synthetic data outperform those trained only on real data for in- and cross-domain generalization. We summarize our contributions as follows:

1. We present the first systematic study of the synthetic-to-real gap of 3D hand pose estimation. Having full control over the rendering pipeline, we establish associations between hand-image components and predictions by decomposing synthetic data.
2. We propose a high-quality synthetic data synthesis pipeline that supports our analysis and enables future applications to boost model performance.
3. We demonstrate that synthetic data can achieve the same level of accuracy as real data while paving the path to use synthetic data alone for hand pose estimation.

2. Related Work

Synthetic Hand Datasets. Building real-world 3D hand datasets is a challenging and time-consuming task. Recording often requires multi-camera systems or depth sensors [15, 33, 51, 56]. After data collection, manual or semi-automated labeling is required for ground truth. Synthetic data is a more time- and cost-effective alternative even if the images are not always photo-realistic; RHD [55], SynthHands [36] and ObMan [17] are three early examples of single hands, ego-centric hands, and hand grasping objects. More recently, DARTset [14] introduced a wrist-enhanced hand model and hands with different accessories. Ego3D [24], Re:InterHand [34] and RenderIH [22] feature two hands interacting, from both ego-centric [24] and third-person views [22, 34]. The datasets differ in their skeleton topology, consideration of background, lighting, and hand texture for rendering (see Tab. 1). The stand-alone efficacy of these synthetic datasets is unestablished, as most are used only for pre-training and then supplemented with either real or mixed data fine-tuning [10, 14, 17, 22, 34]. As we will show, these datasets still exhibit a significant synthetic-to-real gap; our work is the first systematic study to examine this gap for 3D hand pose estimation.

Synthetic-to-Real 3D Hand Pose Estimation. Several works have explored using synthetic data to enhance the generalization performance on real-world data for hand pose estimation. In the depth domain, [31, 44] estimated

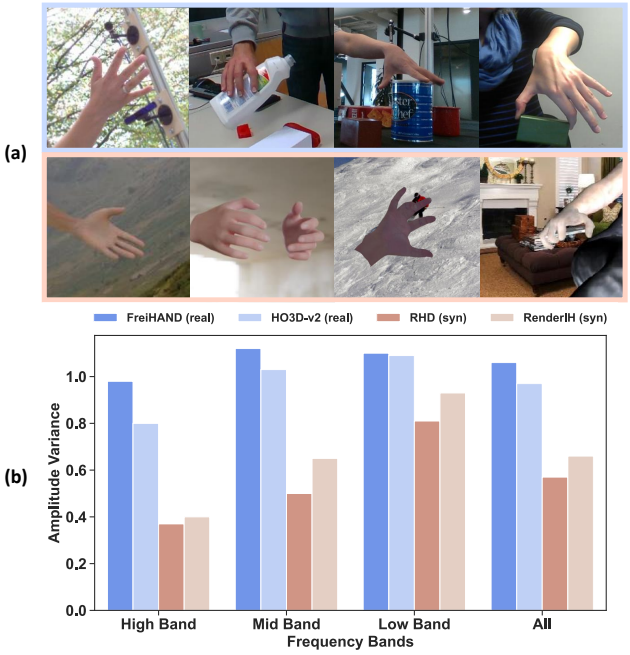

Figure 2. (a) Real images (blue) have richer hand texture information and complex environments than synthetic images (red). Arms and interacting objects are often present in real images. (b) Real datasets have higher variation in amplitude values than synthetic ones across all frequency bands.

hand pose from synthetic depth. For event data, Eventhands [41] used synthetic event streams for training to capture hand poses. Our work focuses on the RGB domain; one of the original works [55] proposes a synthetic RGB dataset (RHD), which reports better generalization results than a limited stereo real hand dataset. Other works learn from both labeled synthetic and unlabeled real-world data, using strategies such as pseudo-labeling and consistency training [50], reconstruction-based approach pseudo-labels [28], multi-modality supervisions [6, 27], input and output representations alignment to build a unified framework for various domain adaptive pose estimation [19].

3. Synthetic-to-Real Hand Analysis

This work verifies the hand image components (Sec. 3.1) that contribute to the synthetic-to-real gap in hand pose estimation. To that end, we develop a hand image synthesis pipeline (Sec. 3.2) for high-quality synthetic images that control the diversity of hand textures, backgrounds, and poses. We apply a compositional strategy (Sec. 3.3) to show the impact of arms and interacting objects. Moreover, we show the connection between occlusion levels and the hand object prior (Sec. 3.4).

3.1. Hand Image Component

A typical hand image consists of the hand, the arm, interacting object(s), and the background. Comparing real hand images [8, 15, 42, 56] with synthetic ones [14, 17, 22, 48, 55], we observe several differences across these components (see Fig. 2(a)). We summarize the differences below.

Appearance. (1) The hands in real images come from multiple subjects and exhibit diverse hand textures; in synthetic images, the textures are limited and unrealistic. (2) Real images often feature complex and diverse environments, whereas synthetic images generally have simple and limited backgrounds. (3) Real images feature hands together with the arms, while synthetic images often only have a floating hand without the arm. (4) PASTA [9] showed that synthetic images have less variance than real images in their high-frequency components for semantic segmentation. We similarly analyze the variance of the amplitude in the frequency spectrum for real versus synthetic hand datasets (see Fig. 2(b)). We find that synthetic hand datasets have less variance than real datasets across the entire frequency spectrum.

Objects. (1) Real images often feature hand-object interactions, leading to various occlusions. (2) Some objects may appear in both real train and test sets, while others are never seen in the synthetic datasets. Therefore, the synthetic datasets may lack the prior information on object occlusion that is present in the real datasets.

Skeleton Topology. The skeletons may differ for real and synthetic hand data. The skeleton topology is a systematic difference between human-annotated and synthetic labels created by parametric or commercial models.

3.2. Hand Image Synthesis

We propose a rendering process (see Fig. 3) starting with a parametric hand model that models bones, muscles, skins, and textures. We then sample hand poses from real-world hand datasets. To push the boundary of realism, each hand is 'dressed up' with high-resolution textures and dynamic backgrounds under varying lighting conditions.

Hand Model. We choose the NIMBLE [23] hand model, which takes pose θ, shape β and texture α parameters as inputs and generates a realistic 3D hand:

$$\mathcal{N}(\theta, \beta, \alpha) = \{\mathcal{G}(\theta, \beta), \mathcal{A}(\alpha)\}. \quad (1)$$

Above, $\mathcal{G}(\theta, \beta)$ denotes the hand geometry, including skin, muscle, and bone meshes. The term $\mathcal{A}(\alpha)$ models the hand texture, including diffuse, specular, and normal maps. Most existing synthetic hand datasets [14, 17, 22] use the

Figure 3. Image synthesis process.

Figure 4. Same hand in different textures.

MANO hand model [40]. NIMBLE has a finer 3D mesh, with 5,990 vertices and 9,984 triangular faces, compared to the 778 vertices and 1,538 faces of MANO. NIMBLE and MANO have different topologies, including differences in both the number of joints (25 vs. 21) and their positions (See Supp. Sec. F).

Hand Pose. We keep our pose distribution consistent with the target real datasets (*i.e.*, FreiHAND [56] and DexYCB [8]) to eliminate its influence during our analysis. We derive the hand pose by fitting the NIMBLE hand mesh to the MANO hand mesh. During the fitting process, we optimize the pose, shape, rotation, and translation parameters in two stages to avoid poor fitting results (See Supp. Sec. B). Since our data synthesis pipeline is controllable and generalizable, we further augment our pose distribution with the fitting algorithm in three ways: (1) sampling poses from ObMan [17] to incorporate more hand-object interaction, (2) randomly synthesizing poses from NIMBLE space, and (3) sampling from VAE prior built with depth-based hand datasets [1].

Hand Texture. We synthesize rich hand textures using the NIMBLE model; it is formulated by $\mathcal{A}(\alpha) = \bar{A} + \Phi\alpha$, where α is the appearance parameter, $\bar{A}$ is the average appearance and Φ are the model principal components. Existing synthetic hand datasets have unrealistic hand textures or limited texture diversity (*e.g.*, RenderIH only offers 30 textures). NIMBLE hand textures are the linear interpolation of 38 photo-realistic hand texture assets from different ages, genders, and races to enhance texture diversity. Some examples are shown in Fig. 4.

Background and Lighting. We use 669 HDRI scenes from [52] as the background; these images feature realistic indoor and outdoor scenes and have been used by [3, 22, 34, 43] to provide highly photo-realistic rendered scenes. Some examples are shown in Fig. 5.

Rendering. We render with Blender Python based on the ray-tracing rendering engine Cycles [4]. First, we import the skin mesh and apply texture to the mesh with diffuse, specular, and normal maps. The camera is set based on the target real datasets' camera intrinsics when analyzing. Backgrounds and textures are selected randomly. This setup creates an automated rendering pipeline contained within Python with minimal effort. It takes around 1 second to render an image with a single RTX A5000 GPU, with further parallelization speedups. More details can be found in Supplementary Sec. C.

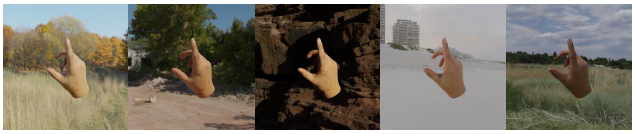

Figure 5. Same hand and hand texture in different backgrounds and lighting. The same hand looks different under different backgrounds and lighting.

Figure 6. Synthetic hand images with realistic hand textures, backgrounds, lighting, arms, and interacting objects.

3.3. Decomposition and Composition

Since our work focuses on analyzing the importance of different components, we segment the arm and object from real images and compose them into synthetic images, as shown in Fig. 1. The composition is more practical and serves as a better control case than rendering these components.

To segment the arm and object from real images, we use the pre-trained object detector Grounding DINO [30] to extract bounding boxes for these components. The bounding box crops are then fed into the segmentation network SAM [21] to estimate the corresponding segmentation masks. For each real image I^i_{real}, where $i \in [1, N]$, N is the number of poses in the target real dataset, we extract the bounding boxes $bbox^i$ of the arm or interacting object with input text prompts x by object detector $D(\cdot)$: $bbox^i = D(x)$. Then the bounding boxes $bbox^i$ are passed to SAM $S(\cdot)$ to obtain the corresponding masks M^i: $M^i = S(bbox^i)$.

The final synthetic image $\tilde{I}^j_{syn}$ with arm and interacting object can be defined as follows:

$$\tilde{I}^j_{\text{syn}} = (1 - M^i_{\text{obj}} - M^i_{\text{arm}}) \odot I^j_{\text{syn}} + M^i_{\text{obj}} \odot I^i_{\text{real}} + M^i_{\text{arm}} \odot I^i_{\text{real}}, \quad (2)$$

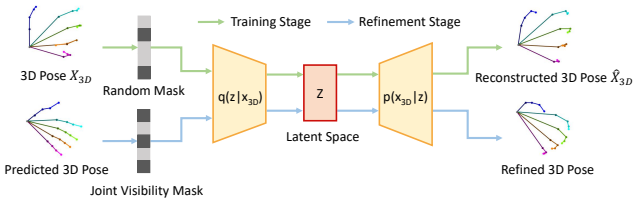

Figure 7. We build a VAE object occlusion prior and use this prior to refine the predicted hand poses.

where I^j_{syn} denotes the rendered synthetic image, M^i_{obj} and M^i_{arm} denote the extracted object mask and arm mask, $\odot$ denotes element-wise multiplication and $j = k * N + i$, where $k \in \mathbb{Z}_{\geq 0}$. In this way, we get the final synthetic images. Examples are shown in Fig. 6.

3.4. Object Occlusion Levels and Priors

To analyze the influence of object occlusion in synthetic-to-real adaptation, we divide the validation set into different levels of occlusion using [47], which provides finger-level occlusion labels (five fingers and palm) for each hand, ranging from 0 (least) to 6 (highest). Moreover, to show the connection between occlusion levels and hand object priors, we employ a VAE prior to reconstruct occluded joints based on visible joints. Similar to [49, 57], we use the VAE [20] to construct the object occlusion prior, which consists of an encoder $q(z|x_{3D})$ and a decoder $p(x_{3D}|z)$ (see Fig. 7). During the training stage, input 3D pose x_{3D} is first randomly masked to increase reconstruction diversity. Then the encoder maps the masked 3D pose x_{3D} to latent space z, and the decoder reconstructs a 3D pose $\hat{x}_{3D}$. A Kullback-Leibler divergence loss and a 3D joint loss are used to train the object occlusion prior:

$$L_{VAE} = \lambda L_{KL} + \|\hat{x}_{3D} - x_{3D}\|^2_2. \quad (3)$$

During the refinement stage, we use the pre-trained VAE prior to refine the predicted joints. More details are provided in Secs. D and E of the Supplementary.

4. Experiments

4.1. Datasets and Metrics

We use FreiHAND and Dex-YCB as the target real datasets and synthesize their corresponding synthetic datasets, SynFrei and SynDex, for analysis.

- **FreiHAND** [56] has 130k training and 4k testing images of a single hand. Each training sample features a real RGB image captured on a green screen background and three extra images with synthetic backgrounds.
- **Dex-YCB** [8] is a large-scale 3D hand-object dataset with over 582k images, featuring objects from the YCB dataset [46]. We use the default "S0" train/test split and a reduced version of the training set, with 35k images for analysis.
- **HO3D-v2** [15] features 3D hand-object interactions, with 66k training and 11k testing images. The ten interacting objects used in the dataset are from the YCB dataset. We use HO3D-v2 to perform cross-dataset validation to evaluate the model's performance on samples with unseen interacting objects.
- **MOW** [7] is a 3D dataset of humans manipulating objects in-the-wild, including 512 annotated samples. It is used as the test set to evaluate the generalization ability.

Evaluation Metrics. We evaluate hand pose and shape accuracy using standard metrics PA-MPJPE and PA-MPVPE in cm, which compare the mean per joint position error and mean per vertex position error of prediction and ground truth after Procrustes Alignment.

4.2. Baselines and Training Details

We reproduce five 3D hand pose estimation methods: S^2HAND [12], CMR [11], METRO [25], MeshGraphormer [26] and simpleHand [54] on the real target datasets as baselines using their official training code. Additionally, we reproduce these methods on our synthesized datasets with amplitude spectrum augmentation [9] (See Supplementary Sec. A) for comparisons to demonstrate the effectiveness of our data synthesis pipeline. All baselines are trained with one NVIDIA RTX A5000 GPU. We mainly use S^2HAND to analyze the synthetic-to-real gap.

4.3. Comparisons with Baselines

We train five methods on the real dataset FreiHAND as baselines and on our synthesized SynFrei to show the effectiveness of our synthetic data synthesis pipeline. As shown in Tab. 2, models trained with only our synthetic dataset can achieve state-of-the-art results, which achieve 97%, 91%, 88%, 91% and 84% of the performance of models trained with the real dataset for S^2HAND, CMR, METRO, MeshGraphormer, and simpleHand respectively. These results indicate a synthetic-to-real gap of around 0.1 cm. We notice decreased performance when training with only synthetic data, especially with stronger model backbones. We speculate that the stronger models over-fit more easily to the synthetic data, affecting the synthetic-to-real generalization performance.

We also train S^2HAND on the reduced version of Dex-YCB and our synthesized SynDex, achieving comparable performance on the Dex-YCB test set of 0.86 cm vs. 0.87 cm in PA-MPJPE.

4.4. Comparison with Other Synthetic Datasets

We compare with DARTset [14], which has a more diverse pose distribution, hand wrists, and accessories but lacks dynamic backgrounds, interacting objects, and amplitude spectrum augmentation. DARTset samples their hand poses from FreiHAND and augments them by linear

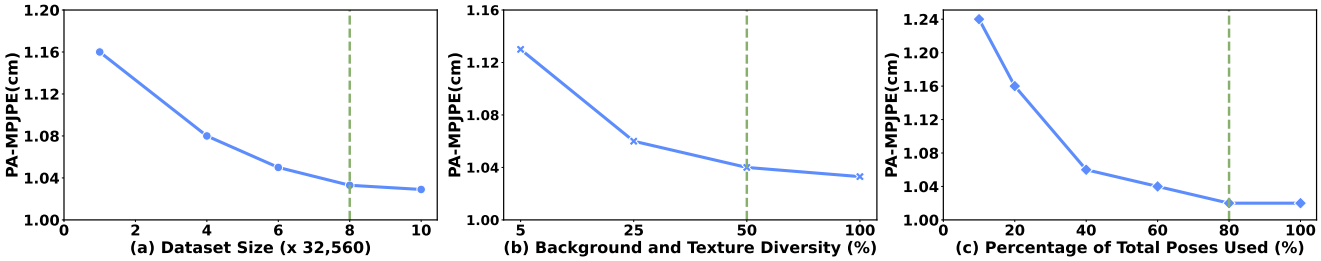

Figure 8. (a) Synthetic dataset size influence. (b) Background and hand texture diversity influence. (c) The percentage of real hand poses used for synthesizing data. The green lines indicate the saturation points.

Table 2. Comparisons of methods trained on FreiHAND and only our SynFrei on the FreiHAND test set (PA-MPJPE↓ / PA-MPVPE↓).

Method	Real FreiHAND	Our SynFrei	Syn→Real
S^2HAND [12]	0.99 / 1.02	1.02 / 1.05	97%
CMR [11]	0.77 / 0.78	0.85 / 0.88	91%
METRO [25]	0.69 / 0.71	0.78 / 0.79	88%
MeshGraphormer [26]	0.69 / 0.70	0.76 / 0.78	91%
simpleHand [54]	0.65 / 0.66	0.77 / 0.79	84%

Table 3. Comparisons of methods trained on DARTset and our SynFrei on the FreiHAND test set (PA-MPJPE↓ / PA-MPVPE↓). The results of DARTset are from its paper.

Method	DARTset	SynFrei
CMR	2.56 / 2.57	**0.85 / 0.88**
METRO	1.77 / 1.75	**0.78 / 0.79**

interpolating the synthesized pose from A-MANO space and FreiHAND sample pose. As shown in Tab. 3, CMR and METRO trained with our SynFrei outperform DARTset despite our dataset size being half DARTset (325,600 vs. 758,378). Moreover, CMR trained with our SynDex outperforms DARTset on the FreiHand test set (1.92 / 1.98), even with different pose distributions.

Comparisons between single- and two-hand datasets are challenging due to the domain gap. We still evaluate the cross-dataset generalization on the HO3D-v2 test set for models trained with SynFrei and RenderIH [22]. We only take the right-hand results of the RenderIH model. Models trained with SynFrei achieve better cross-dataset evaluation than RenderIH of 1.41 cm vs. 1.76 cm in PA-MPJPE. This result can be attributed to the limited expressiveness of interacting hands, which fails to let the model learn hand-object interactions, and the blurry background effects in RenderIH images due to the low background resolution.

4.5. Appearance Analysis

Dataset Size and Diversity Influence. We analyze the influence of the synthetic dataset size on synthetic-to-real performance. As illustrated in Fig. 8(a) and, as expected, there are diminishing returns with adding more data. When SynFrei's synthetic dataset size reaches 8×32,560, performance gains plateau and the trend of error decrease becomes flat.

Table 4. Arm, amplitude spectrum augmentation, and interacting object influence on FreiHAND and Dex-YCB test sets (PA-MPJPE↓). "Amp Aug" is short for amplitude spectrum augmentation. ✗ refers to assigning RGB values to the arm and object mask positions.

	Components			Train set	
	Arm	Amp Aug	Object	SynFrei	SynDex
(i)		✓	✓	1.07	0.90
(ii)	✓		✓	1.11	0.89
(iii)	✓	✓		1.07	0.95
(iv)	✗	✓	✗	1.04	0.92
(v)	✓	✓	✓	**1.02**	**0.87**

We further render synthetic datasets at this full size of 8×32,560 samples, but with varying levels of diversity: 50%, 25%, and 5%, where each value indicates the percentage of the full background and hand texture assets. As shown in Fig. 8(b), more diverse backgrounds and hand textures can reduce the synthetic-to-real gap. This result suggests that enriching the background and texture assets is useful when synthesizing data. However, the performance plateaus at 50%, indicating that around 300 indoor and outdoor HDRI scenes are sufficient for synthesizing hand data. Several techniques can be implemented in our synthesis pipeline to further improve performance, such as changing camera viewpoints, flipping or rotating backgrounds, and adjusting skin tone contrast, brightness, or subsurface color.

Arm Influence. We analyze the importance of the arm by removing it from synthetic images. Tab. 4 shows increases in PA-MPJPE for models trained with SynFrei and SynDex without the arm component comparing Rows (i) and (v). Fig. 9(a) shows that the model trained on SynFrei without the arm tends to misinterpret parts of the arm as the wrist, especially if the arm occupies a large portion of the image. Such errors do not occur for SynFrei with arms. Since the

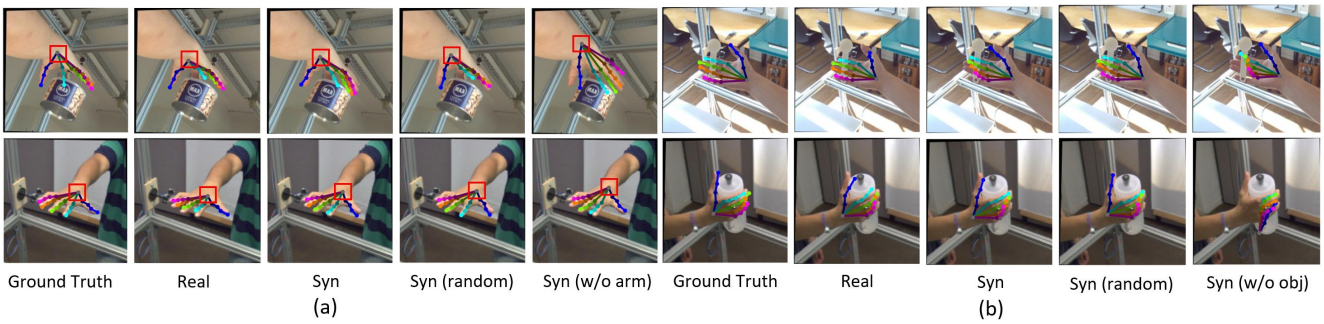

Figure 9. (a) Comparisons of model trained with FreiHAND (Real), our SynFrei (Syn), our SynFrei with random arm (random), and our SynFrei (w/o arm) on FreiHAND test set. (b) Comparisons of model trained with FreiHAND, SynFrei, SynFrei with random interacting object and SynFrei (w/o object) on the FreiHAND test set.

arm and hand share similar colors, models can easily be confused and incorrectly locate the root joint on the arm rather than the wrist if synthetic images lack arm information. This result suggests that the arm is an important cue for locating the wrist, thus affecting the overall hand pose.

Amplitude Spectrum Augmentation Influence. We find that amplitude spectrum augmentation is a crucial component. Removing it results in increases in PA-MPJPE for both SynFrei (0.09 cm) and SynDex (0.02 cm) (see Tab. 4 Rows (ii) and (v)). This suggests that the ability of amplitude spectrum augmentation to distort amplitude information while preserving phase information (*e.g.*, hand structure) can enhance model robustness when the domain shifts from synthetic to real.

Random Arm and Object Occlusion Influence. We further randomize the arm and object incorporated into the hand image by assigning RGB values to the corresponding arm and object mask positions described in Sec. 3.3. For the arm, we assign it to the mean RGB value of the hand to better imitate hand-arm compatibility. For the object occlusion, we assign a random RGB value to the object position. Tab. 4 Rows (iv) and (v) show that with random arm and object occlusion, although the synthetic-to-real gaps slightly increase compared with segmenting from real images, the synthetic-to-real performance can still achieve 95% and 93% on FreiHAND and Dex-YCB test sets. This result reveals that the model can learn how to locate the wrist position and interacting objects accurately without highly realistic arm or object occlusion information (see Fig. 9).

4.6. Pose Analysis

We analyze the influence of hand pose on the synthetic-to-real performance by controlling the number of real hand poses used for data synthesis. Note that we still diversify each hand with different backgrounds and textures for rendering. As shown in Fig. 8(c), the synthetic-to-real performance improves when synthetic hand poses approach the real ones. This result suggests that the pose difference between the synthetic and real datasets introduces a synthetic-to-real gap. We also observe that the model trained with synthetic data can achieve around 80% of the final performance using only 5% of total real poses. With 20% of real poses, the performance increases to around 90%, and with 40% of real poses, it reaches 97%. In other words, synthetic hand images effectively enable the model to learn visual representations, while core hand poses are sufficient to achieve good generalization performance.

4.7. Object Occlusion Analysis

Tab. 4 Rows (iii) and (v) show increases in PA-MPJPE after removing the interacting objects from SynFrei (0.05 cm) and SynDex (0.08 cm), especially for the Dex-YCB test set, which features more hand-object interactions. When the object largely occludes the hand, the predicted hand pose does not interact with the object right after removing interacting objects from synthetic images (see Fig. 9(b)).

Occlusion Levels. We use the HO3D-v2 dataset for cross-dataset validation, which features hand-object interactions. Since HO3D-v2 test set labels are not publicly available, we separate the original train set into train and validation sets.

We compare models' performance on validation sets under three training settings: S^2HAND trained with HO3D-v2 train set (S_1), FreiHAND (S_2), and SynFrei (S_3). As shown in Fig. 10(a), a performance gap exists between S_1 and S_2, S_3. As the occlusion level rises, S_2 and S_3 degrade due to their lack of object occlusion information in the validation sets, while S_1 improves, which incorporates this prior.

We further investigate the distribution of unique object amount across each occlusion level and discover a relationship between the performance of S_1 and the unique object amount. As shown in Fig. 10(b), S_1 performs the best on occlusion level 6, which has the least amount of unique objects. With object occlusion prior information in validation sets, S_1 can associate hand poses with specific objects. Also, higher occlusion levels are often associated

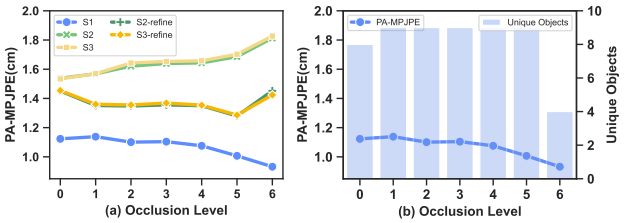

Figure 10. (a) Comparisons of S_1, S_2 and S_3 on validation sets with different occlusion levels. S_2 and S_3 after object occlusion prior refinement are denoted as S_2-refine and S_3-refine. (b) Relationship between the performance of S_1 and the number of unique objects at each occlusion level.

with larger objects, which impose greater constraints on hand poses and result in smaller variations in poses. Therefore, S_1 demonstrates improved performance on higher occlusion levels. However, without object occlusion information, the performances of S_2 and S_3 decline as the occlusion levels increase. These findings show that object occlusion prior contributes to the synthetic-to-real gap.

Object Occlusion Prior. We use Dex-YCB to train a VAE object occlusion prior (See Sec. 3.4), which includes the objects in HO3D-v2. As shown in Fig. 10(a), the synthetic-to-real performance of S_3 improves after applying the VAE object occlusion prior to refine the predicted joints (S_3-refine). Incorporating the object occlusion prior bridges the gap between synthetic and real hand datasets with high levels of occlusion. While previous works use prior information to refine poses in real-to-real scenarios [37, 39, 57], our findings also show its utility in synthetic-to-real adaptation.

Through our object occlusion analysis, we provide a potential training scheme to better adapt synthetic data to real applications. Models can initially be trained on large-scale synthetic data to learn hand representations. Existing hand-object interaction datasets can then be used to provide useful prior information to refine the predictions and further reduce the synthetic-to-real gap.

4.8. Hand Skeleton Topology Analysis

Different hand skeleton topologies can have different joint definitions [13]. This can introduce a synthetic-to-real gap since the synthetic data's skeleton topology may differ from real datasets. To analyze the influence of hand skeleton topology difference, we train S^{2}HAND without label adaption and obtain NIMBLE joints as the ground-truth joints by fitting NIMBLE mesh to MANO mesh for training. Note that the NIMBLE skeleton differs from the MANO skeleton (see Supp. F), and we remove the extra anatomical joints of NIMBLE when evaluating. As shown in Tab. 5, training with NIMBLE joints and without label adaption, the synthetic-to-real error increases from 1.02 cm to 1.28 cm. This result demonstrates that differences in hand skeleton topology between synthetic and real datasets can introduce bias to the prediction.

Table 5. Hand skeleton topology differences.

Label Adaptation	PA-MPJPE↓
-	1.28
✓	**1.02**

Table 6. Cross-dataset generalization results in PA-MPJPE ↓. †: results come from the official evaluation server.

Train Set \ Test Set	FreiHAND	Dex-YCB	HO3D-v2†	RHD	MOW
FreiHAND	0.99	1.14	1.39	1.40	1.47
+ SynFrei	**0.95**	**1.11**	**1.28**	**1.32**	**1.38**
Dex-YCB	1.63	0.86	1.25	1.72	1.24
+ SynDex	**1.49**	**0.82**	**1.15**	**1.65**	**1.18**

4.9. Cross-Dataset Evaluation

We demonstrate the ability of our synthetic data to bridge the gap across different hand datasets through cross-dataset generalization using S^{2}HAND. As shown in Tab. 6, the model trained with a mixture of real and synthetic data achieves better results than only real data for in- and cross-domain generalization. For example, the model trained with Dex-YCB and SynDex performs better than with only Dex-YCB on the FreiHAND test set. The difference may be because Dex-YCB mostly consists of indoor images, while the FreiHAND test set contains many outdoor images. Incorporating synthetic images with both indoor and outdoor environments bridges this gap. These results prove our synthetic data is a good complement to real data, both in- and cross-domain. More results can be found in Supplementary Sec. H.

5. Conclusion

This paper proposes the first systematic study of the synthetic-to-real gap of 3D hand pose estimation. We analyze the synthetic-to-real gap with respect to appearance, hand pose, object occlusion, and hand skeleton topology. We impressively demonstrate that synthetic hand data can achieve the same level of accuracy as real hand data while paving the path to use synthetic data alone. Models trained on a mixture of real and our synthetic data outperform those trained only on real data for in-domain and cross-domain generalization. Our proposed data synthesis pipeline can create high-quality synthetic hand images to alleviate our reliance on real data and benefit future research and applications.

Acknowledgements

This research / project is supported by the Ministry of Education, Singapore, under the Academic Research Fund Tier 1 (FY2022). Additionally, this work is supported by the National Natural Science Foundation of China (No. 62406298), the Guangzhou Municipal Nansha District Science and Technology Bureau (No.2022ZD012), the HSBC project L0562, and the Fundamental Research Funds for the Central Universities, Communication University of China (CUC24XT07).

References

[1] Anil Armagan, Guillermo Garcia-Hernando, Seungryul Baek, Shreyas Hampali, Mahdi Rad, Zhaohui Zhang, Shipeng Xie, MingXiu Chen, Boshen Zhang, Fu Xiong, et al. Measuring generalisation to unseen viewpoints, articulations, shapes and objects for 3d hand pose estimation under hand-object interaction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 4

[2] Gwangbin Bae, Martin de La Gorce, Tadas Baltrušaitis, Charlie Hewitt, Dong Chen, Julien Valentin, Roberto Cipolla, and Jingjing Shen. Digiface-1m: 1 million digital face images for face recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3526–3535, 2023. 1

[3] Michael J Black, Priyanka Patel, Joachim Tesch, and Jinlong Yang. Bedlam: A synthetic dataset of bodies exhibiting detailed lifelike animated motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8726–8737, 2023. 1, 4

[4] Blender Foundation. Cycles Renderer. https://www.cycles-renderer.org/, 2021. Accessed on 5th August, 2023. 4

[5] Giuseppe Caggianese, Nicola Capece, Ugo Erra, Luigi Gallo, and Michele Rinaldi. Freehand-steering locomotion techniques for immersive virtual environments: A comparative evaluation. *International Journal of Human–Computer Interaction*, 36(18):1734–1755, 2020. 1

[6] Yujun Cai, Liuhao Ge, Jianfei Cai, and Junsong Yuan. Weakly-supervised 3d hand pose estimation from monocular rgb images. In *Proceedings of the European conference on computer vision (ECCV)*, pages 666–682, 2018. 3

[7] Zhe Cao, Ilija Radosavovic, Angjoo Kanazawa, and Jitendra Malik. Reconstructing hand-object interactions in the wild. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12417–12426, 2021. 5

[8] Yu-Wei Chao, Wei Yang, Yu Xiang, Pavlo Molchanov, Ankur Handa, Jonathan Tremblay, Yashraj S Narang, Karl Van Wyk, Umar Iqbal, Stan 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*, pages 9044–9053, 2021. 3, 4, 5

[9] Prithvijit Chattopadhyay, Kartik Sarangmath, Vivek Vijaykumar, and Judy Hoffman. Pasta: Proportional amplitude spectrum training augmentation for syn-to-real domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19288–19300, 2023. 2, 3, 5

[10] Liangjian Chen, Shih-Yao Lin, Yusheng Xie, Yen-Yu Lin, and Xiaohui Xie. Mvhm: A large-scale multi-view hand mesh benchmark for accurate 3d hand pose estimation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 836–845, 2021. 1, 2

[11] Xingyu Chen, Yufeng Liu, Chongyang Ma, Jianlong Chang, Huayan Wang, Tian Chen, Xiaoyan Guo, Pengfei Wan, and Wen Zheng. Camera-space hand mesh recovery via semantic aggregationand adaptive 2d-1d registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1, 5, 6

[12] Yujin Chen, Zhigang Tu, Di Kang, Linchao Bao, Ying Zhang, Xuefei Zhe, Ruizhi Chen, and Junsong Yuan. Model-based 3d hand reconstruction via self-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10451–10460, 2021. 5, 6

[13] Xiaoming Deng, Yuying Zhu, Yinda Zhang, Zhaopeng Cui, Ping Tan, Wentian Qu, Cuixia Ma, and Hongan Wang. Weakly supervised learning for single depth-based hand shape recovery. *IEEE Transactions on Image Processing*, 30:532–545, 2021. 2, 8

[14] Daiheng Gao, Yuliang Xiu, Kailin Li, Lixin Yang, Feng Wang, Peng Zhang, Bang Zhang, Cewu Lu, and Ping Tan. Dart: Articulated hand model with diverse accessories and rich textures. *Advances in Neural Information Processing Systems*, 35:37055–37067, 2022. 1, 2, 3, 5

[15] Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. Honnotate: A method for 3d annotation of hand and object poses. In *CVPR*, 2020. 2, 3, 5

[16] Shangchen Han, Po-chen Wu, Yubo Zhang, Beibei Liu, Linguang Zhang, Zheng Wang, Weiguang Si, Peizhao Zhang, Yujun Cai, Tomas Hodan, et al. Umetrack: Unified multi-view end-to-end hand tracking for vr. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 1

[17] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11807–11816, 2019. 1, 2, 3, 4

[18] Songyao Jiang, Bin Sun, Lichen Wang, Yue Bai, Kunpeng Li, and Yun Fu. Skeleton aware multi-modal sign language recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3413–3423, 2021. 1

[19] Donghyun Kim, Kaihong Wang, Kate Saenko, Margrit Betke, and Stan Sclaroff. A unified framework for domain adaptive pose estimation. In *European Conference on Computer Vision*, pages 603–620. Springer, 2022. 3

[20] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 5

[21] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 4

[22] Lijun Li, Linrui Tian, Xindi Zhang, Qi Wang, Bang Zhang, Liefeng Bo, Mengyuan Liu, and Chen Chen. Renderih: A large-scale synthetic dataset for 3d interacting hand pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20395–20405, 2023. 1, 2, 3, 4, 6

[23] Yuwei Li, Longwen Zhang, Zesong Qiu, Yingwenqi Jiang, Nianyi Li, Yuexin Ma, Yuyao Zhang, Lan Xu, and Jingyi Yu. Nimble: a non-rigid hand model with bones and muscles. *ACM Transactions on Graphics (TOG)*, 41(4):1–16, 2022. 3, 2

[24] Fanqing Lin, Connor Wilhelm, and Tony Martinez. Two-hand global 3d pose estimation using monocular rgb. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2373–2381, 2021. 2

[25] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *CVPR*, 2021. 1, 5, 6

[26] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12939–12948, 2021. 5, 6

[27] Qiuxia Lin, Linlin Yang, and Angela Yao. Cross-domain 3d hand pose estimation with dual modalities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17184–17193, 2023. 3

[28] Qiuxia Lin, Kerui Gu, Linlin Yang, and Angela Yao. Synthetic-to-real pose estimation with geometric reconstruction. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[29] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pages 740–755. Springer, 2014. 4

[30] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 4

[31] Jameel Malik, Ahmed Elhayek, Fabrizio Nunnari, Kiran Varanasi, Kiarash Tamaddon, Alexis Heloir, and Didier Stricker. Deephps: End-to-end estimation of 3d hand pose and shape by learning from synthetic depth. In *2018 International Conference on 3D Vision (3DV)*, pages 110–119, 2018. 2

[32] Gyeongsik Moon and Kyoung Mu Lee. I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 752–768. Springer, 2020. 1

[33] Gyeongsik Moon, Shoou-I Yu, He Wen, Takaaki Shiratori, and Kyoung Mu Lee. Interhand2.6m: A dataset and baseline for 3d interacting hand pose estimation from a single rgb image. In *European Conference on Computer Vision (ECCV)*, 2020. 2

[34] Gyeongsik Moon, Shunsuke Saito, Weipeng Xu, Rohan Joshi, Julia Buffalini, Harley Bellan, Nicholas Rosen, Jesse Richardson, Mize Mallorie, Philippe Bree, Tomas Simon, Bo Peng, Shubham Garg, Kevyn McPhail, and Takaaki Shiratori. A dataset of relighted 3D interacting hands. In *NeurIPS Track on Datasets and Benchmarks*, 2023. 1, 2, 4

[35] Gyeongsik Moon, Shunsuke Saito, Weipeng Xu, Rohan Joshi, Julia Buffalini, Harley Bellan, Nicholas Rosen, Jesse Richardson, Mallorie Mize, Philippe De Bree, et al. A dataset of relighted 3d interacting hands. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[36] Franziska Mueller, Dushyant Mehta, Oleksandr Sotnychenko, Srinath Sridhar, Dan Casas, and Christian Theobalt. Real-time hand tracking under occlusion from an egocentric rgb-d sensor. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2017. 2

[37] Hyeongjin Nam, Daniel Sungho Jung, Yeonguk Oh, and Kyoung Mu Lee. Cyclic test-time adaptation on monocular video for 3d human mesh reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14829–14839, 2023. 8

[38] Junho Park, Kyeongbo Kong, and Suk-Ju Kang. Attention-hand: Text-driven controllable hand image generation for 3d hand reconstruction in the wild. In *European Conference on Computer Vision*, pages 329–345. Springer, 2024. 4

[39] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J Guibas. Humor: 3d human motion model for robust pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11488–11499, 2021. 8

[40] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *arXiv preprint arXiv:2201.02610*, 2022. 4, 2

[41] Viktor Rudnev, Vladislav Golyanik, Jiayi Wang, Hans-Peter Seidel, Franziska Mueller, Mohamed Elgharib, and Christian Theobalt. Eventhands: Real-time neural 3d hand pose estimation from an event stream. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12385–12395, 2021. 3

[42] Srinath Sridhar, Franziska Mueller, Michael Zollhoefer, Dan Casas, Antti Oulasvirta, and Christian Theobalt. Real-time joint tracking of a hand manipulating an object from rgb-d input. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2016. 3

[43] Tze Ho Elden Tse, Franziska Mueller, Zhengyang Shen, Danhang Tang, Thabo Beeler, Mingsong Dou, Yinda Zhang, Sasa Petrovic, Hyung Jin Chang, Jonathan Taylor, et al. Spectral graphormer: Spectral graph-based transformer for egocentric two-hand reconstruction using multi-view color images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14666–14677, 2023. 4

[44] Chengde Wan, Thomas Probst, Luc Van Gool, and Angela Yao. Self-supervised 3d hand pose estimation through training by fitting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[45] Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Sebastian Dziadzio, Thomas J Cashman, and Jamie Shotton. Fake it till you make it: face analysis in the wild using synthetic data alone. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3681–3691, 2021. 1

[46] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017. 5

[47] Hao Xu, Tianyu Wang, Xiao Tang, and Chi-Wing Fu. H2onet: Hand-occlusion-and-orientation-aware network for real-time 3d hand mesh reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17048–17058, 2023. 5, 3

[48] Linlin Yang and Angela Yao. Disentangling latent hands for image synthesis and pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9877–9886, 2019. 3

[49] Linlin Yang, Shile Li, Dongheui Lee, and Angela Yao. Aligning latent spaces for 3d hand pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2335–2343, 2019. 5

[50] Linlin Yang, Shicheng Chen, and Angela Yao. Semihand: Semi-supervised hand pose estimation with consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11364–11373, 2021. 1, 3

[51] Lixin Yang, Kailin Li, Xinyu Zhan, Fei Wu, Anran Xu, Liu Liu, and Cewu Lu. OakInk: A large-scale knowledge repository for understanding hand-object interaction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[52] Greg Zaal, S. Majboroda, and A. Mischok. HDRI Haven. https://hdrihaven.com, 2023. Accessed on 5th August, 2023. 4

[53] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023. 4

[54] Zhishan Zhou, Shihao Zhou, Zhi Lv, Minqiang Zou, Yao Tang, and Jiajun Liang. A simple baseline for efficient hand mesh reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1367–1376, 2024. 5, 6

[55] Christian Zimmermann and Thomas Brox. Learning to estimate 3d hand pose from single rgb images. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 2, 3

[56] Christian Zimmermann, Duygu Ceylan, Jimei Yang, Bryan Russell, Max Argus, and Thomas Brox. Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 813–822, 2019. 2, 3, 4, 5

[57] Binghui Zuo, Zimeng Zhao, Wenqian Sun, Wei Xie, Zhou Xue, and Yangang Wang. Reconstructing interacting hands with interaction prior from monocular images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9054–9064, 2023. 5, 8

Analyzing the Synthetic-to-Real Domain Gap in 3D Hand Pose Estimation

Supplementary Material

In the following supplementary material, we present more details about our work. Sec. A provides the amplitude spectrum augmentation algorithm used during the model training stage to augment synthetic images. Sec. B provides the fitting algorithm for obtaining NIMBLE hand meshes in different poses. Sec. C provides the rendering details of the hand image synthesis pipeline. Sec. D illustrates how we obtain the finger-level occlusion annotations for object occlusion analysis. Sec. E provides the implementation details of the VAE object occlusion prior. Sec. F shows the difference between NIMBLE and MANO hand templates. Sec. G compares the influence of hand model templates used for data synthesis. Sec. H presents the results of training the model with a combination of synthetic and real data. Sec. I shows the component influence before Procrustes Alignment. Sec. J shows the comparison of synthetic data generated by generative models. Sec. K shows more visualization of our synthetic dataset.

A. Amplitude Spectrum Augmentation

We apply amplitude spectrum augmentation [9] during the model training stage to address the frequency domain gap. The amplitude spectrum augmentation algorithm is shown in Alg. 1. We set the hyper-parameters α to 3, β to 0.25, and k to 2 during the training.

Algorithm 1 Amplitude spectrum augmentation

Input: $x \in \mathbb{R}^{H \times W}$; ▷ Input synthetic image
Augmentation parameters: α, k, β
$\mathcal{F}(x) \leftarrow \text{FFT}(x)$; ▷ Fast Fourier Transform
$\mathcal{A}(x), \mathcal{P}(x) \leftarrow \text{Abs}(\mathcal{F}(x)), \text{Ang}(\mathcal{F}(x))$
$\mathcal{A}(x) \leftarrow \text{FFTShift}(\mathcal{A}(x))$; ▷ Zero-center amplitude
for $p \in [-H/2, H/2]$ **do**
  for $q \in [-W/2, W/2]$ **do**
    $\sigma[p, q] \leftarrow \left(2\alpha\sqrt{\frac{p^2+q^2}{H^2+W^2}}\right)^k + \beta$
  end for
end for
$\lambda \sim \mathcal{N}(1, \sigma^2)$; ▷ Calculate perturbations
$\tilde{\mathcal{A}}(x) \leftarrow \lambda \odot \mathcal{A}(x)$; ▷ Perturb amplitude spectrum
$\tilde{\mathcal{A}}(x) \leftarrow \text{FFTShift}(\tilde{\mathcal{A}}(x))$
$\tilde{x} \leftarrow \text{Inverse-FFT}(\tilde{\mathcal{A}}(x), \mathcal{P}(x))$; ▷ Augmented image

B. Fitting Algorithm

To prepare hand meshes for rendering, we develop algorithms to fit NIMBLE [23] meshes to MANO [40] meshes.

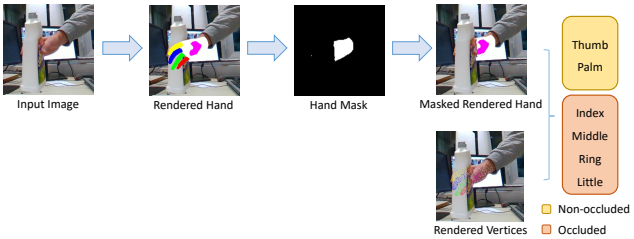

Figure 1. Finger-level occlusion annotation preparation.

Since we find that optimizing all parameters directly leads to poor fitting results, we divide the fitting process into two stages. In the coarse stage, we optimize the rotation parameter $r \in \mathbb{R}^3$ and translation parameter $t \in \mathbb{R}^3$. The training loss consists of a 3D joint loss and a vertex loss:

$$L = L_{joint} + \lambda_{vert} L_{vert},$$

where λ_{vert} is set to 0.1. We use a learning rate of 1e0 for both r and t. The epoch for the coarse stage is 2, with 3,000 iterations per epoch.

In the fine stage, we optimize pose $\theta \in \mathbb{R}^{30}$, shape $\beta \in \mathbb{R}^{20}$, rotation r and translation t parameters together. The training loss consists of a vertex loss and regularization losses for pose and shape parameters:

$$L = L_{vert} + \lambda_{pose} L_{pose} + \lambda_{shape} L_{shape},$$

where λ_{pose} and λ_{shape} are set to 50. We use a learning rate of 1e-3 for both θ and β, and a learning rate of 1e-2 for both r and t. The epoch for the fine stage is 4, with 3,000 iterations per epoch. All learning rates are divided by 10 every 1,000 iterations. Additionally, learning rates are reset to their initial value at the beginning of each epoch to prevent convergence to local minima.

C. Rendering Details

In this section, we detail our camera settings for rendering. To ensure our analysis is not affected by other influencing factors, our camera viewpoints are consistent with the target real dataset we compared. We calculate the camera focal length f^i for rendering each synthetic image i by:

$$f^i = f_x^i * \frac{W_{sensor}}{W_{image}},$$

where f_x^i is derived from the provided camera intrinsic matrix K^i, W_{sensor} represents the sensor width, which is 36 mm, and W_{image} represents the width of the rendered image in pixel, which can be 224 or 256. Multi-view settings can also be implemented using our rendering code.

Encoder	
Linear(512)	ReLU
Linear(512)	ReLU
Linear(512)	ReLU
Linear(512)	

Table A. VAE encoder.

Decoder	
Linear(512)	ReLU
Linear(512)	ReLU
Linear(512)	ReLU
Linear(512)	ReLU
Linear(512)	ReLU
Linear(512)	

Table B. VAE decoder.

D. Occlusion Level Division

We introduce how we collect the finger-level occlusion annotations using [47]. As shown in Fig. 1, we first allocate faces and vertices of five fingers and the palm with different colors and then render the hand and vertices into the input image. Since HO3D-v2 [15] provides the hand mask for each image, we apply this mask to remove the rendered hand parts occluded by the object. With the masked rendered hand and rendered vertices, we can compare each vertex color against the corresponding finger color to calculate the number of occluded vertices. Here, we set the threshold of occluded vertices to 40 to determine whether the finger can be considered occluded. With these occlusion annotations, we can divide the validation set into different occlusion levels and perform cross-dataset validation.

E. VAE Architecture

The architecture of VAE object occlusion prior with an encoder and a decoder consists of a series of (Linear, ReLU) layers (see Tab. A and Tab. B). The dimension of latent variable z is set to 64. The batch size is 128 and the learning rate is 1e-4. The total epoch for training the VAE object occlusion prior is 160. λ is set to 0.01. In the training stage, a Kullback-Leibler divergence loss and a 3D joint loss are used to train the object occlusion prior and the random mask is applied to the input 3D pose. In the refinement stage, we apply the visibility mask to the predicted 3D pose and use the object occlusion prior to reconstruct those occluded joints. The visibility masks are obtained through a render-and-compare method similar to the finger-level occlusion annotations [47].

F. Hand Skeleton Topology Difference

The hand skeleton topology difference between NIMBLE and MANO is shown in Fig. 2. Different hand skeleton topologies can have different joint rig definitions and this difference can introduce bias to synthetic-to-real adaptation.

G. NIMBLE vs. MANO

We render 10k synthetic images in NIMBLE and MANO models respectively. We keep other influencing factors consistent, such as poses and background diversity. The MANO texture maps are collected from RenderIH [22]. As shown in Tab. C, using NIMBLE achieves better performance than MANO across all metrics. This suggests that leveraging a finer hand mesh template for image synthesis is beneficial for bridging the synthetic-to-real gap.

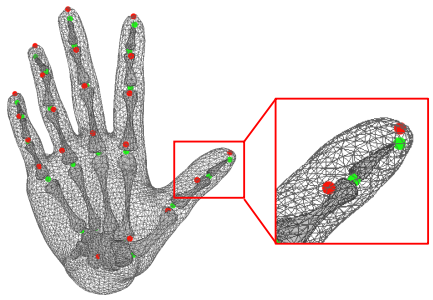

Figure 2. Hand skeleton topology difference between NIMBLE template and MANO template. The green dot denotes the NIMBLE joints and the red dot denotes the MANO joints.

Table C. Comparison of S^2HAND trained with 10k synthetic images in MANO and NIMBLE template and tested on FreiHAND

Template	PA-MPJPE ↓	PA-MPVPE ↓	MPJPE ↓	MPVPE ↓
MANO	1.29	1.31	3.79	3.95
NIMBLE	**1.24**	**1.26**	**2.95**	**3.06**

H. Synthetic-to-Real Adaptation

We explore the advantages of pre-training on our total synthetic data when targeting FreiHAND and Dex-YCB. As shown in Fig. 3, by leveraging our synthetic data for pre-training, the model achieves good performance even when fine-tuned with only a small fraction of real data.

Furthermore, in the in-the-wild scenario, synthetic data is useful as well. As shown in Tab. D, complementing real datasets with our total synthetic data achieves better performance on MOW. A qualitative comparison is presented in Fig. 4.

Table D. Cross-dataset evaluation of S^2HAND on in-the-wild dataset MOW.

Train sets	PA-MPJPE ↓	Train sets	PA-MPJPE ↓
FreiHAND	1.47	Dex-YCB	1.24
+ SynFrei	1.38	+ SynDex	1.18
+ Total Syn Data	**1.19**	+ Total Syn Data	**1.15**

I. Component Influence before Procrustes Alignment

In this section, we show how each component influences the final predictions in metrics before Procrustes Alignment

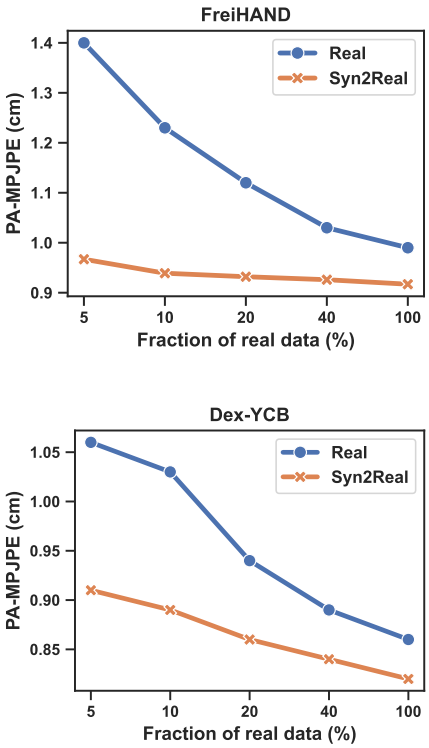

Figure 3. Comparison of training with real data only and pre-training with total synthetic data followed by fine-tuning on real data (Syn2Real).

Figure 4. Qualitative comparison on MOW.

Table E. Before vs. after Procrustes Alignment on FreiHAND test set across different factors. ✗ refers to assigning RGB values to the arm and object mask positions.

Arm	Amp Aug	Object	PA-MPJPE ↓		MPJPE ↓	
✓	✓	✓	1.02		1.99	
	✓	✓	1.07	**-0.05 (-5%)**	2.22	**-0.23 (-12%)**
✓		✓	1.11	**-0.09 (-9%)**	2.42	**-0.43 (-22%)**
✓	✓		1.07	**-0.05 (-5%)**	2.13	**-0.14 (-7%)**
✗	✓	✗	1.04	**-0.02 (-2%)**	2.07	**-0.08 (-4%)**

(MPJPE). As shown in Tab. E, adding different components has greater impacts on MPJPE than PA-MPJPE. For example, the improvement in MPJPE is greater than PA-MPJPE (12% vs. 5%) for the arm. This confirms arm is important to locate the wrist and global rotation. Moreover, amplitude spectrum augmentation is crucial to reduce the synthetic-to-real gap. Removing it during training leads to a 22% performance decline in MPJPE.

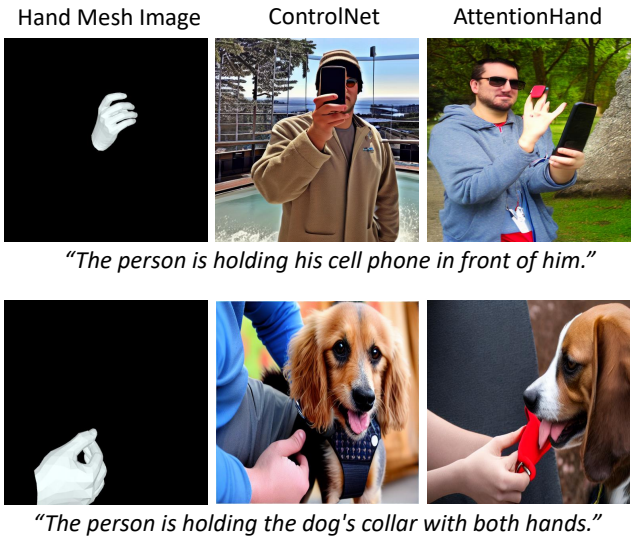

Figure 5. Synthetic hand data generated by text-to-image generative models with the hand mesh image control and text prompt.

J. Comparison with Data from Generative Models

We run the official code of ControlNet [53] and AttentionHand [38] to generate synthetic hand data given mesh images and text prompts. Since the pre-trained checkpoint is not open-sourced by AttentionHand currently, we train it from scratch. We also train the ControlNet given hand mesh images and text prompts as conditions. MSCOCO [29] is used as the training set following AttentionHand. As shown in Fig. 5, although the generated hand images have realistic backgrounds, the hands do not align well to the given mesh prompts. While generative models have significant potential for data synthesis, further research is needed to explore how to produce large-scale, plausible, and reliable synthetic data effectively.

K. Additional Visualizations

Fig. 6 and Fig. 7 show some examples of synthetic hand images and synthetic hand images with arms and interacting objects.

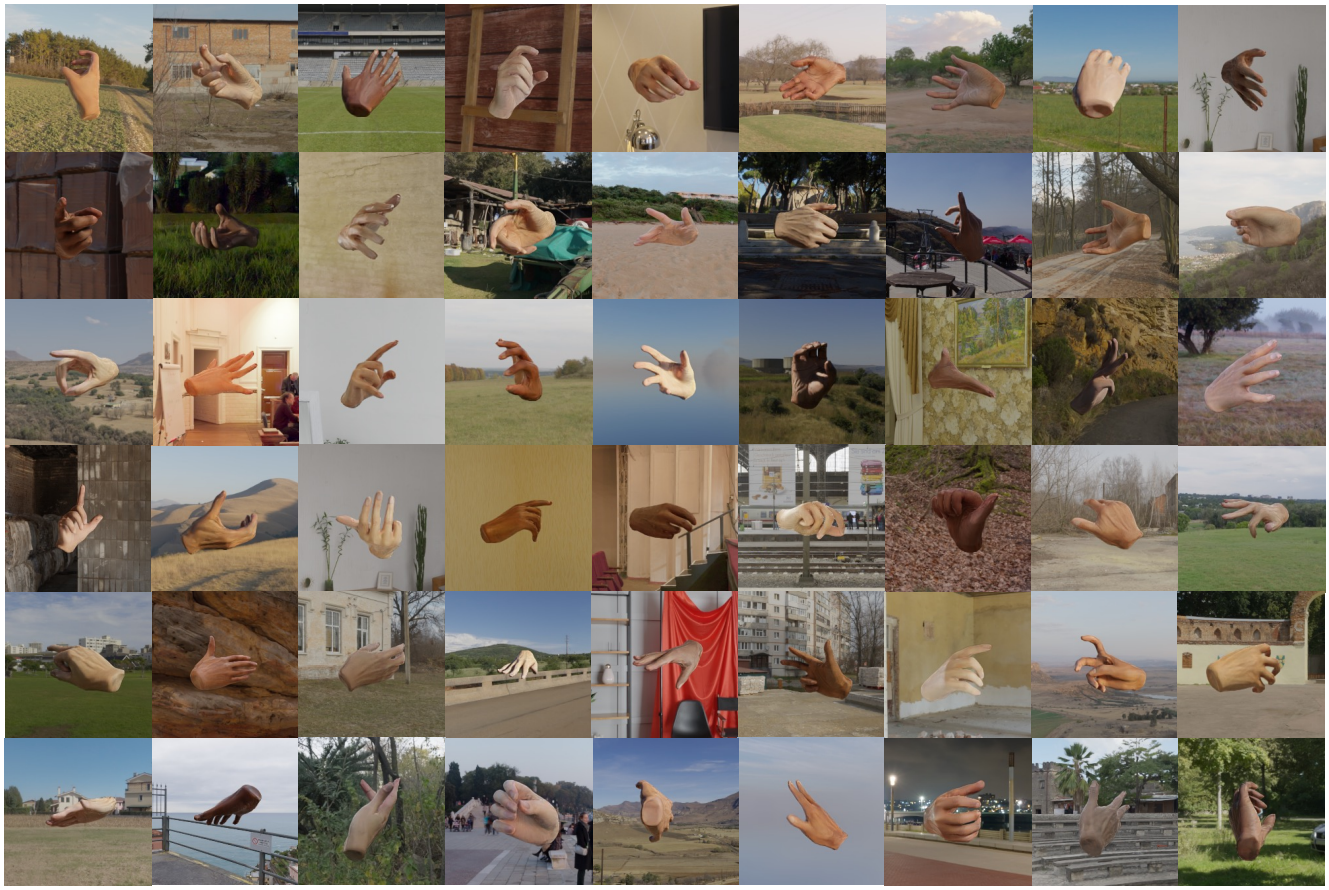

Figure 6. Synthetic hand image samples in our dataset.

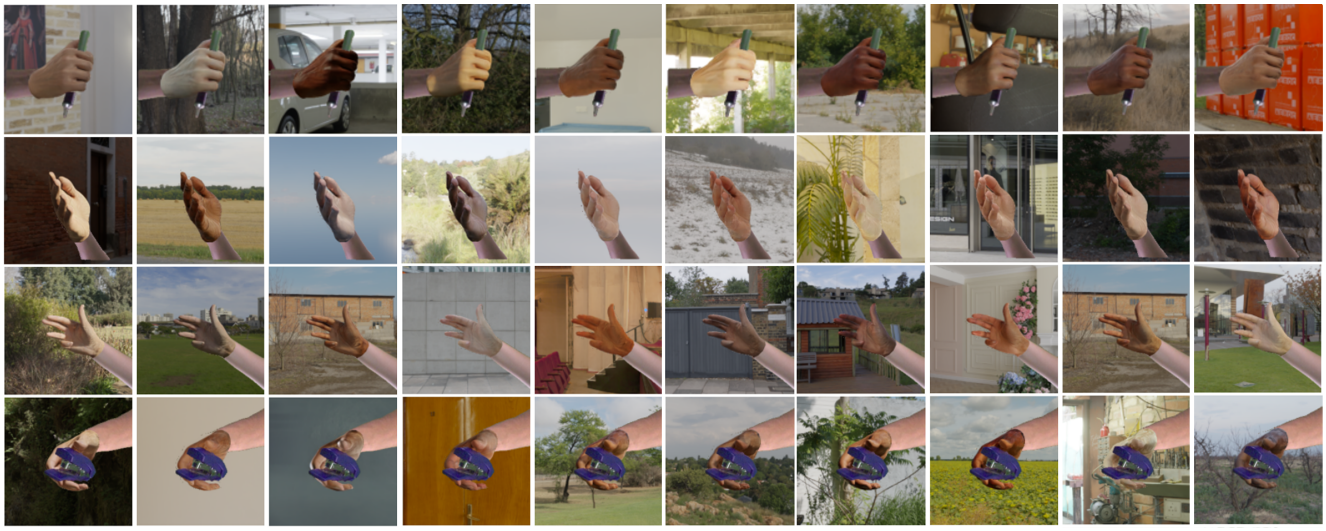

Figure 7. Synthetic hand images with arms or interacting objects.