arXiv:2510.00527v2 [cs.CV] 6 Sep 2026

Cascaded Diffusion Framework for Probabilistic Coarse-to-Fine Hand Pose Estimation

Taeyun Woo, Jinah Park, and Tae-Kyun Kim

KAIST
{taeyun.woo, jinahpark, kimtaekyun}@kaist.ac.kr

Abstract. Deterministic models for 3D hand pose reconstruction, whether single-staged or cascaded, struggle with pose ambiguities caused by self-occlusions and complex hand articulations. Existing cascaded approaches progressively refine pose predictions in a coarse-to-fine manner, but their deterministic nature prevents them from modeling pose uncertainty. Conversely, recent probabilistic methods capture pose distributions but are confined to single-stage estimation, often yielding inaccurate 3D reconstructions without refinement. To address these limitations, we propose a coarse-to-fine cascaded diffusion framework that combines probabilistic modeling with cascaded refinement. The first stage is a joint diffusion model that samples diverse 3D joint hypotheses, and the second stage is a Mesh Latent Diffusion Model (Mesh LDM) that reconstructs a 3D hand mesh conditioned on a joint sample. Our key idea is to use these diverse hypotheses as a training signal rather than as final outputs, so that the Mesh LDM learns distribution-aware joint–mesh relationships and becomes robust to the variation in coarse predictions. Extensive ablations validate the necessity of the cascaded design and the choice of latent-space diffusion. Experiments on FreiHAND, HO3Dv2, and DexYCB show that our method achieves state-of-the-art performance and remains stable under occlusion and pose ambiguity.

Keywords: Hand Pose Estimation · Diffusion Models · Cascaded Framework · Probabilistic Estimation

1 Introduction

3D hand pose estimation (HPE) is an important research area for emerging applications such as VR/AR [19, 62], robotics [22, 39, 44], and human-computer interaction [60, 61]. HPE aims to predict hand poses from images, and numerous approaches [6–8, 12, 13, 38, 41, 42, 47, 52, 65] have achieved promising results. Despite recent progresses, HPE remains a challenging task due to self-occlusions, complex hand articulation, and diverse hand shapes.

Traditional HPE approaches [23, 41, 75] typically adopt a single-stage pipeline that directly regresses 3D joint positions or parameterized hand models such as MANO [55] from input images. However, by estimating 3D hand poses in a single step, these methods lack a refinement process that can progressively resolve

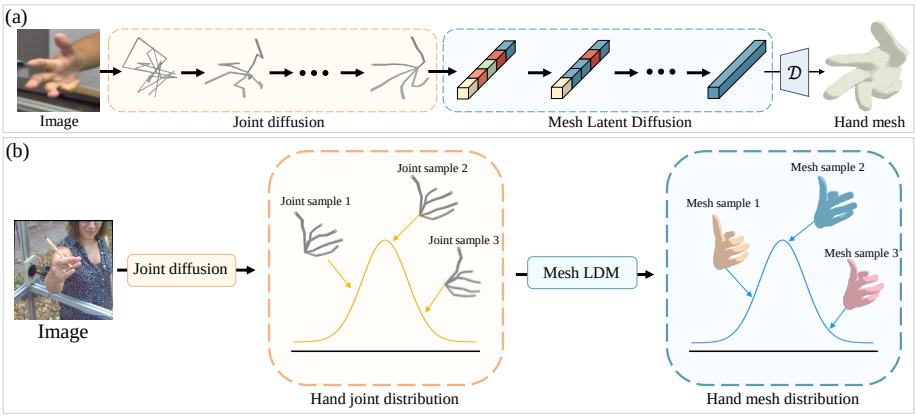

Fig. 1: A cascaded diffusion model for hand pose estimation. (a) The joint diffusion model first denoises noise into 3D hand keypoints, and the Mesh Latent Diffusion Model (Mesh LDM) then reconstructs the hand mesh from them. (b) Rather than conditioning on a single keypoint estimate, the Mesh LDM is trained on diverse joint samples drawn from a pose distribution, so it learns distribution-aware joint–mesh relationships.

pose ambiguities. As a result, learning the highly non-linear mapping from 2D observations to 3D hand configurations becomes particularly challenging under occlusions and complex articulations.

To address these issues, cascaded architectures [7,12,47,52,65] have been proposed, where an initial hand pose estimation (*e.g.*, joint positions or MANO parameters) is progressively refined through subsequent stages. By decoupling HPE into coarse estimation and fine-level refinement, cascaded models have shown improved performance compared to single-stage models. However, these methods remain deterministic, producing only a single prediction and failing to capture the uncertainty and diversity of valid hand poses, which limits their ability to handle the pose ambiguities.

Recently, denoising diffusion models [25, 54] emerged as powerful generative frameworks for modeling complex data distributions. They have shown remarkable success in image generation [53], 3D object generation [31], and human motion synthesis [66]. Beyond generative tasks, diffusion models also show potential in human pose estimation [14, 17, 27, 59] and hand pose estimation [9, 28, 38], leveraging the generative ability to model pose distributions. These models are capable of sampling diverse pose hypotheses from complex distributions, however, existing methods adopt the single-stage design, which limits their ability to refine noisy predictions or recover fine-grained details.

In this paper, we propose a novel cascaded diffusion framework for 3D hand pose estimation (Fig. 1) that combines the coarse-to-fine cascaded framework with the probabilistic modeling power of diffusion models. Specifically, our method consists of two stages: a joint diffusion model that denoises 3D keypoints con-

ditioned on 2D input, and a mesh latent diffusion model (Mesh LDM) that reconstructs the 3D mesh latent vector conditioned on the denoised joint and image features. Our key idea is to use the diverse joint hypotheses from the first stage as a condition for training the second stage, rather than as final outputs. This allows the second stage to learn from the distribution of plausible joints rather than a single deterministic estimate, improving robustness when the coarse prediction is uncertain.

Unlike previous diffusion-based methods that operate in 3D space [38] or MANO space [10], our approach performs diffusion in a latent space. As illustrated in Fig. 2, diffusion in (a) 3D space often leads to the loss of surface geometry and pose structure. By contrast, (b) MANO space leverages a hand shape prior with predefined hand parts and can better preserve hand shapes, but it still tends to lose joint articulation fidelity. In (c), diffusion in the latent space preserves both pose structure and surface details, resulting in more expressive and realistic hand representations.

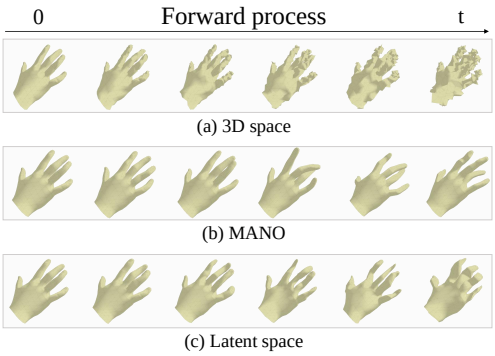

Fig. 2: Forward diffusion process in different spaces. The figure shows how the noise progressively changes the 3D hand mesh across different representations.

Furthermore, since the mesh diffusion model is conditioned on a joint sample drawn from the joint diffusion model, it learns mesh distributions over a range of plausible poses rather than a single set of deterministic keypoints. This design allows the model to capture distribution-aware joint-mesh relationships, improving robustness and performance in challenging scenarios.

Our contributions are summarized as follows:

- We propose a cascaded diffusion framework for hand pose estimation that couples a probabilistic joint stage with a mesh refinement stage in a coarse-to-fine manner.
- By conditioning the mesh diffusion model on a joint sample from diverse joint hypotheses generated from joint diffusion model, our method learns distribution-aware joint–mesh relationships. This improves accuracy and robustness to pose ambiguity.
- Our method achieves state-of-the-art performance on DexYCB [5] and competitive results on FreiHAND [75] and HO3Dv2 [20] with extensive experiments validating its effectiveness.

2 Related works

2.1 3D Hand Pose Estimation

3D hand pose estimation (HPE) has been extensively studied, with numerous approaches proposed over the past decade [1–4, 6–8, 11–13, 16, 32, 33, 35, 36, 38, 41, 42, 47, 52, 65, 67, 73]. One common paradigm in this field [2, 4, 7, 13, 23, 49, 52, 73] involves predicting a parameterized hand model, such as MANO [55], from an input image. These methods leverage prior knowledge of the hand model for robust estimation but are inherently limited by the predefined hand shape and lack fine-grained pose variations. Alternatively, other approaches [12, 16, 32, 38, 41, 47] predict hand keypoints and mesh vertices directly without predefined priors, offering greater flexibility. However, these single stage approaches often suffer from geometric inconsistencies and noisy predictions.

To improve robustness and accuracy, cascaded models [7, 12, 47, 52] decompose the HPE pipeline into a coarse-to-fine two-stage model. The first stage predicts an initial pose (*e.g.*, 3D keypoints or MANO parameters), while the second stage refines the prediction by incorporating additional details. However, existing deterministic cascaded models fail to model pose uncertainty, particularly in occluded or ambiguous scenarios. This limitation motivates the exploration of stochastic approaches, such as diffusion models, for HPE.

2.2 Diffusion-Based Pose Estimation

Denoising diffusion models [25, 54] have demonstrated remarkable success in generative tasks, such as image synthesis [53, 54], 3D object generation [31, 34], and human motion synthesis [40, 66]. Their ability to capture complex distributions has recently been explored in human pose estimation [10, 14, 15, 17, 27, 46, 59]. However, their application to hand poses remains largely unexplored, with only a few recent works [9, 28, 38].

Existing diffusion-based hand pose estimating methods typically adopt a single-stage approach, directly estimating either hand keypoints [9, 28] or meshes [38]. While diffusion models effectively model complex distributions, these methods lack a structured refinement process, limiting their robustness under occlusions and complex articulations. This motivates our cascaded diffusion framework, which introduces a coarse-to-fine refinement process into diffusion-based hand estimation.

2.3 Cascaded Diffusion Models

While traditional diffusion models operate in a single-stage fashion, recent works have introduced cascaded diffusion frameworks across tasks such as image generation [26, 37, 57], 3D object generation [31, 34], and video synthesis [29, 72]. These models progressively refine outputs by conditioning each stage on the results of the previous stage.

In 3D domains, InterHandGen [34] generates two-hands poses in two steps, and SALAD [31] decomposes 3D object synthesis into multiple representations. While InterHandGen generates interaction scenarios and SALAD focuses on generating static objects, our method estimates articulated 3D hand poses from an image. Unlike these generation tasks, where diverse outputs are the goal, our cascade serves accuracy: the diverse hypotheses of the first stage are a means to train a more robust refinement stage.

The key advantage of cascaded diffusion in regression tasks is that the second stage is conditioned on a sample drawn from a distribution of plausible outputs, rather than a single deterministic one. This enables greater robustness by leveraging the stochastic nature of diffusion models, allowing better handling of ambiguities and pose uncertainty. Our proposed cascaded diffusion framework extends this idea to hand pose estimation, ensuring that the generated samples represent the distribution of 3D hand poses.

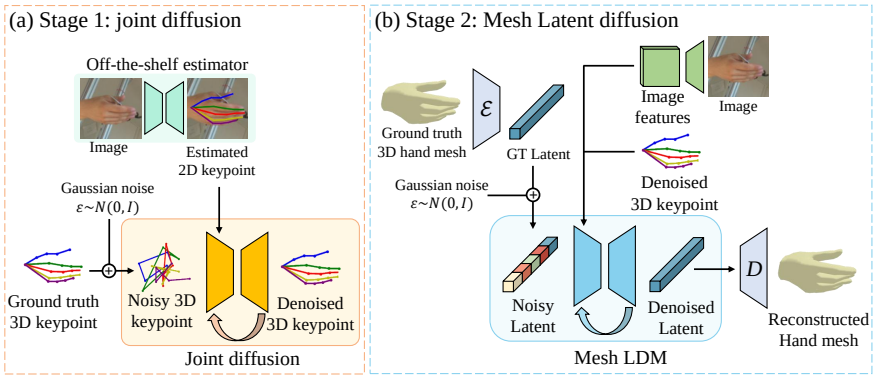

Fig. 3: Overview of the proposed cascaded diffusion model. (a) The joint diffusion model lifts 2D keypoints from an off-the-shelf estimator into 3D joints. (b) With the joint model fixed, the Mesh LDM denoises the hand-mesh latent, conditioned on a denoised 3D joint and image features. The latent is decoded into the final mesh by the pre-trained Mesh AE decoder.

3 Method

As shown in Fig. 3, we propose a coarse-to-fine cascaded diffusion framework for 3D hand pose estimation. Our model consists of two stages: a joint diffusion model that estimates 3D hand joints from 2D keypoints, and a mesh latent diffusion model (Mesh LDM) that reconstructs the 3D hand mesh from the denoised joint and image features. This probabilistic coarse-to-fine design allows the model to represent multiple plausible joint hypotheses instead of a single deterministic prediction, and refine the mesh conditioned on the learned joint distribution.

3.1 Background: Denoising Diffusion Model

Denoising diffusion models [25, 54] generate samples through a forward process that gradually corrupts data with Gaussian noise and a reverse process that learns to denoise it. The forward process follows a Markov chain $q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\,\mathbf{x}_{t-1}, \beta_t \mathbf{I})$, where β_t is the noise variance at step t, and the reverse process $p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t)$ is learned by the model. Following previous pose estimation approaches [38, 58, 66], we adopt the $\mathbf{x}_0$-parameterization, training the model to directly reconstruct the clean data $\mathbf{x}_0$ rather than the added noise with following denoising loss: $\mathcal{L}_{DDPM} = \mathbb{E}_{\mathbf{x}_0,t}\left[\|\mathbf{x}_0 - \hat{\mathbf{x}}_0\|^2\right]$.

3.2 Stage 1: Joint Diffusion Model

The joint diffusion model lifts 2D hand keypoints into 3D joints. It adapts D3DP [59], originally proposed for 2D-to-3D human pose sequence uplifting, to the hand domain. Given a noisy joint J_t, the model denoises it into a clean 3D joint $\hat{J}_0$, conditioned on 2D keypoints c_{2D}.

Training. The model takes three inputs: a timestep $t \sim \mathcal{U}(0, T)$, 2D keypoints c_{2D} from an off-the-shelf estimator [52], and a noisy joint J_t. The joint diffusion model is trained to directly reconstruct the clean joint by minimizing the diffusion loss $\|J_0 - \hat{J}_0\|^2$.

Inference. Starting from Gaussian noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$, the model progressively denoises it into a 3D joint $\hat{J}_0$ conditioned on c_{2D}. Each generated joint serves as a condition for the Mesh LDM.

3.3 Stage 2: Mesh Latent Diffusion Model

Mesh AutoEncoder. To embed the hand mesh into a latent space, we train a Mesh AutoEncoder (Mesh AE) based on SpiralNet++ [18]. The encoder $\mathcal{E}$ maps a hand mesh $V \in \mathbb{R}^{778\times 3}$ to a latent vector $x \in \mathbb{R}^{168}$, and the decoder $\mathcal{D}$ reconstructs the mesh from it. Mesh AE is trained with vertex and joint reconstruction losses together with a KL term that regularizes the latent toward a Gaussian. Here the joint is obtained from the mesh via the MANO joint regressor [55].

Mesh Latent Diffusion. Mesh LDM p_ϕ denoises a noisy latent x_t into the clean latent x_0, conditioned on the joint sample $\hat{J}_0$ from the joint diffusion model and on image features $\mathcal{I}$. The denoised latent is then decoded into the hand mesh, $\hat{V}_0 = \mathcal{D}(\hat{x}_0)$.

Mesh LDM is built on the DiT framework [50] and operates on a sequence of 21 tokens, one per hand joint. To form the input, the noisy latent is broadcast across the 21 positions and concatenated along the channel axis with the denoised joint $\hat{J}_0$, so that every token carries the full mesh latent together with the coordinate of its corresponding joint. The resulting sequence is projected to the hidden dimension of the transformer.

Image features are extracted as a four-level feature pyramid, and each level is projected to the hidden dimension and flattened into a set of spatial tokens.

Within each block, the tokens are first refined by self-attention; they then attend to the four feature levels through parallel cross-attentions, whose outputs are concatenated and fused by an MLP. The diffusion timestep modulates every block through adaptive layer normalization [51]. After the final block, the tokens are projected and flattened back to the denoised latent $\hat{x}_0$. The overall structure is illustrated in Fig. 4.

3.4 Cascaded Diffusion Framework

Our cascaded framework couples the two stages so that mesh reconstruction is guided by the joint distribution rather than a single deterministic pose. Instead of regressing the mesh directly from the image, we decompose the task into 3D joint estimation followed by mesh reconstruction: the joint diffusion model provides diverse joint hypotheses, and the Mesh LDM reconstructs a mesh conditioned on them.

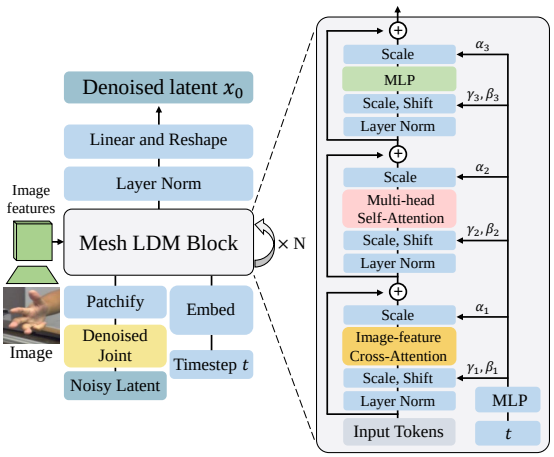

Fig. 4: Mesh LDM architecture. The latent input and denoised joint are processed through DiT blocks [50] with cross-attention to image features.

Training. We first train the joint diffusion model p_θ to lift 2D keypoints into 3D joints. Its weights are then fixed, and the Mesh LDM is trained to reconstruct a mesh latent conditioned on a single joint sample drawn from p_θ, together with image features $\mathcal{I}$. Since p_θ produces different hypotheses at each step, the Mesh LDM is exposed to a range of plausible joint samples over training. This lets it learn a joint–mesh mapping that is robust to the variation in coarse predictions, rather than overfitting to a single deterministic input.

Inference. At inference, the joint diffusion model generates multiple joint hypotheses, which are averaged into a stable estimate. Conditioned on this aggregated joint and image features, the Mesh LDM generates several latent samples that are averaged and decoded into the final mesh. Aggregating at both stages turns the diversity seen during training into a stable point estimate at test time.

3.5 Loss Functions

We employ three loss terms: a diffusion loss, a mesh vertex loss, and a joint loss. Since the joint diffusion model and Mesh AE are already trained, their parameters are fixed while training the Mesh LDM.

The diffusion loss supervises the Mesh LDM to reconstruct the clean latent under the $\mathbf{x}_0$-parameterization, $\mathcal{L}_{DDPM} = \|x_0 - \hat{x}_0\|^2$. The vertex and joint losses enforce accurate geometry and pose consistency:

$$\mathcal{L}_V = \left\|V - \hat{V}\right\|_1, \qquad \mathcal{L}_J = \left\|J - \mathcal{J}\hat{V}\right\|_1, \tag{1}$$

where $\mathcal{J}$ is the joint regressor that extracts 3D joints from the mesh vertices. The total training loss is

$$\mathcal{L} = \lambda_{DDPM}\mathcal{L}_{DDPM} + \lambda_V\mathcal{L}_V + \lambda_J\mathcal{L}_J, \tag{2}$$

where we set $\lambda_{DDPM} = 1$, $\lambda_V = 10$, and $\lambda_J = 5$.

4 Experiments

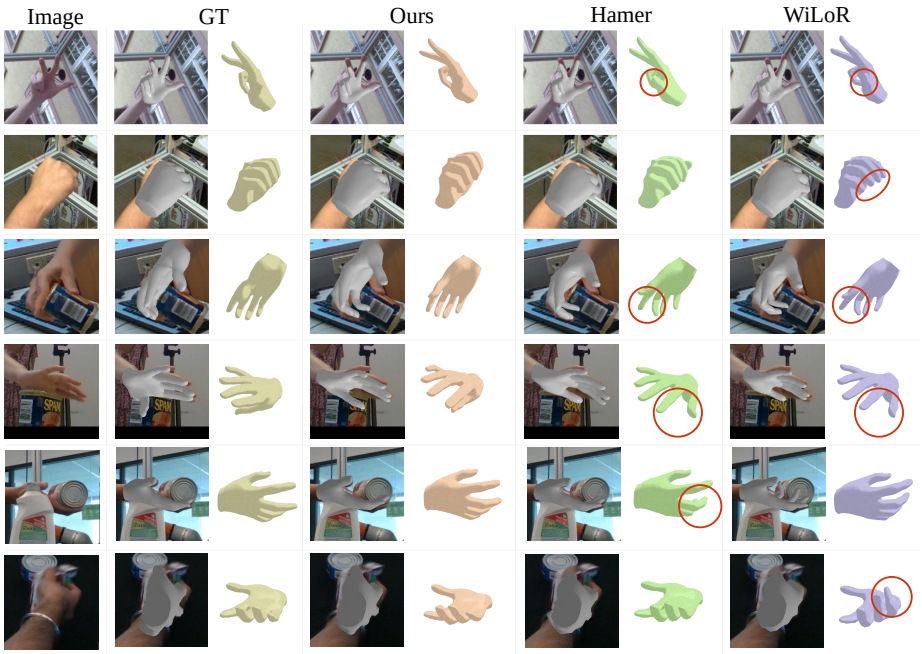

Fig. 5: Qualitative results on FreiHAND [75], HO3Dv2 [20] and DexYCB [5].

4.1 Experimental settings

Implementation details. Our framework is implemented in PyTorch. The joint diffusion model is based on D3DP [59], which uplifts human pose sequences to 3D using the MixSTE backbone [71]; we set the input sequence length to 1 for single-frame estimation. The Mesh AutoEncoder is trained on each dataset

Method	P-MPJPE ↓	P-MPVPE ↓	F@5 ↑	F@15 ↑
I2L-MeshNet [47]	7.4	7.6	0.681	0.973
Pose2Mesh [12]	7.7	7.8	0.674	0.969
I2UV-HandNet [6]	6.7	6.9	0.707	0.977
METRO [41]	6.5	6.3	0.731	0.984
Tang *et al.* [65]	6.7	6.7	0.724	0.981
Lin *et al.* [42]	5.9	6.0	0.764	0.986
MobRecon [7]	5.7	5.8	0.784	0.986
AMVUR [30]	6.2	6.1	0.767	0.987
HaMeR [49]	6.0	5.7	0.785	0.990
Hamba† [13]	5.8	5.5	0.798	0.991
WiLoR† [52]	5.5	**5.1**	**0.825**	**0.993**
HandOS† [8]	**5.0**	5.3	0.812	0.991
HHMR [38]	5.8	5.8	-	-
HHMR (best) [38]	5.3	5.4	-	-
Proposed	**5.0**	5.2	0.816	0.992
Proposed (best)	4.2	4.5	0.866	0.995

Table 1: Comparison with SOTAs on FreiHAND dataset. We evaluate on the standard protocol and report metrics for predicted 3D joint and 3D mesh on FreiHAND. † stands for using additional datasets. Methods above the line are deterministic methods, the others are probabilistic methods. "best" denotes an oracle best-of-32 selection.

separately. Both the joint diffusion model and Mesh LDM use 1000 denoising steps for training, and DDIM sampling [63] with a step size of 10 at inference. Further details are provided in the Supplementary Material.

Datasets. We evaluate on three widely used hand pose benchmarks that span increasing levels of difficulty: FreiHAND [75], HO3Dv2 [20], and DexYCB [5]. FreiHAND is a single-hand dataset with 133K training and 3.9K evaluation images, containing relatively little occlusion. HO3Dv2 and DexYCB are hand-object interaction datasets that involve object-induced occlusion: HO3Dv2 provides 66K training and 11K test images, while DexYCB is substantially larger, with 465K training and 116K test images. We follow the official split for all three datasets.

Training details. We train our cascaded diffusion model separately on each dataset using the AdamW optimizer [45] on a single NVIDIA RTX 4090 GPU with a mini-batch size of 32. The joint diffusion model is trained for 250K iterations and the Mesh LDM for 100K iterations. The initial learning rate is 1e-4 and decays by a factor of 0.9 every 5K iterations under a step-based scheduler.

Evaluation metrics. Following standard evaluation protocols, we assess performances using Procrustes Aligned Mean Per Joint Position Error (P-MPJPE) and Procrustes Aligned Mean Per Vertex Position Error (P-MPVPE). Additionally, we report the fraction of poses with errors below 5mm (F@5) and 15mm (F@15). For DexYCB we additionally report the Area Under the Curve (AUC) for 3D hand joints following HandOS [8].

Method	P-MPJPE ↓	P-MPVPE ↓	F@5 ↑	F@15 ↑
Hasson *et al.* [23]	11.0	11.2	0.464	0.939
Hampali *et al.* [20]	10.7	10.6	0.506	0.942
I2L-MeshNet [47]	11.2	13.0	0.409	0.932
Pose2Mesh [12]	12.5	12.7	0.441	0.909
Liu *et al.* [43]	9.9	9.5	0.528	0.956
I2UV-HandNet [6]	9.9	10.1	0.500	0.943
METRO [41]	10.4	11.1	0.484	0.946
ArtiBoost [70]	11.4	10.9	0.488	0.944
MobRecon [7]	9.2	9.4	0.538	0.957
Keypoint Trans. [21]	10.8	-	-	-
HandOccNet [48]	9.1	8.8	0.564	0.963
AMVUR [30]	8.3	8.2	0.608	0.965
HaMeR [49]	7.7	7.9	0.635	0.980
Hamba† [13]	**7.5**	7.7	**0.648**	0.982
WiLoR† [52]	**7.5**	7.7	0.646	**0.983**
Proposed	**7.5**	**7.5**	0.633	0.982
Proposed (best)	7.4	7.4	0.639	0.982

Table 2: Comparison with the state-of-the-art on the HO3Dv2 dataset. We evaluate on the standard protocol and report metrics for predicted 3D joint and 3D mesh on HO3Dv2. † stands for using additional datasets. "best" denotes an oracle best-of-32 selection.

4.2 Hand Pose Estimation

We evaluate our cascaded diffusion framework on FreiHAND [75], HO3Dv2 [20], and DexYCB [5] through both quantitative and qualitative experiments. As described in Sec. 3, our inference pipeline generates multiple hypotheses at each stage: the joint diffusion model samples 50 joint hypotheses that are averaged into a stable estimate, and the Mesh LDM takes the aggregated joint and the input image to generate 50 latent vectors, whose average is decoded into the final hand mesh.

Quantitative Results. Tab. 1, 2, and 3 report the quantitative results on the three datasets. Our method achieves competitive performance across all three benchmarks, with the best results on DexYCB. Notably, recent methods such as Hamba [13], WiLoR [52], and HandOS [8] are trained on multiple datasets, whereas our model is trained only on the target dataset, indicating strong generalization under limited-data conditions. On the hand-object datasets, HO3Dv2 and DexYCB, where object-induced occlusion is common, our method remains competitive with the best approaches.

Qualitative Results. Fig. 5 presents qualitative comparisons on the three datasets. While all methods produce visually plausible meshes when rendered onto the image, the baselines tend to yield less natural finger articulation when viewed from the side. For example, in the second row, where the ground truth is a *rock* gesture, both HaMeR and WiLoR show slightly awkward finger bend-

Method	P-MPJPE ↓	P-MPVPE ↓	AUC_J ↑
Spurr *et al.* [64]	6.8	–	0.864
MobRecon [7]	6.4	5.6	–
HandOccNet [48]	5.8	5.5	–
H2ONet [68]	5.7	5.5	–
Zhou *et al.* [74]	5.5	5.5	–
WiLoR [52]	4.9	4.8	0.902
HandOS [8]	5.2	5.0	0.896
Proposed	**4.5**	**4.5**	**0.911**
Proposed (best)	4.2	4.2	0.917

Table 3: Comparison with the state-of-the-art on the DexYCB dataset. "best" denotes an oracle best-of-32 selection.

ing that deviates from natural hand priors, whereas our method produces more realistic articulation by leveraging learned pose priors. Similar improvements are observed in other challenging examples, highlighting the advantage of our cascaded framework in producing more natural hand poses.

Best-of-*N* evaluation. We further examine whether the diverse joint hypotheses contain high-quality candidates through a best-of-N evaluation. We sample N joint hypotheses and reconstruct a mesh from each without averaging. We then report the accuracy of the mesh closest to the ground truth. In the main comparisons (Tab. 1, 2, and 3), the "Proposed (best)" rows report the best of 32 hypotheses selected against ground truth, and are therefore oracle upper bounds. Under this setting, our method surpasses HHMR [38], a probabilistic method that diffuses directly in 3D space, on FreiHAND (Tab. 1), where both methods select the best among 32 samples. This indicates that our cascaded latent design refines diverse hypotheses more effectively than single-stage 3D diffusion. The best results nonetheless verify a premise of our design: the hypotheses from the joint stage are not arbitrary noise but spread meaningfully around plausible poses, containing candidates close to the ground truth.

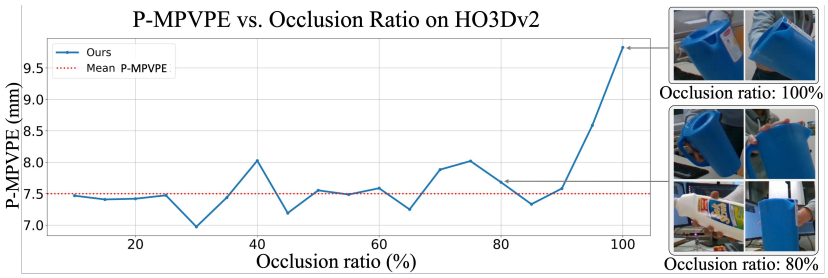

Fig. 6: Robustness to occlusion on HO3Dv2. P-MPVPE as a function of the occlusion ratio, defined as the fraction of the hand region overlapped by the manipulated object. Accuracy remains stable up to roughly 80% occlusion and degrades only under extreme occlusion.

# samples	FreiHAND		HO3Dv2		DexYCB	
	Average	Best	Average	Best	Average	Best
1	5.04	5.04	7.90	7.90	7.61	7.61
10	5.01	4.78	7.89	7.80	7.60	7.40
50	5.01	4.67	7.89	7.75	7.59	7.30
100	5.01	4.63	7.89	7.74	7.59	7.26
200	5.01	4.59	7.89	7.72	7.59	7.23

Table 4: Effect of the number of samples from the joint diffusion model. P-MPJPE (mm, ↓) of the Stage-1 joint output on FreiHAND, HO3Dv2 and DexYCB. "Best" is the oracle sample closest to ground truth.

4.3 Robustness to Occlusion

Hand-object interaction introduces object-induced occlusion, which makes both 2D-to-3D joint lifting and mesh reconstruction substantially harder. To examine how our framework behaves as occlusion becomes more severe, we evaluate on HO3Dv2 [20] while varying the occlusion ratio, defined as the fraction of the hand region overlapped by the manipulated object. We group the test samples by their occlusion ratio and report P-MPVPE for each group.

As shown in Fig. 6, our method maintains stable accuracy up to roughly 80% occlusion, with a sharper drop only under extreme occlusion where little visual evidence remains. We attribute this stability to our training scheme. The Mesh LDM is trained on diverse joint samples drawn from the joint diffusion model, so it learns to reconstruct meshes across a range of plausible joint conditions rather than from a single fixed input. When occlusion makes the coarse joint estimate less reliable, the mesh stage therefore still produces a plausible reconstruction. This robustness follows directly from our core design of using joint diversity as a training signal, rather than being an incidental property.

4.4 Ablation study

To analyze the contribution of each design choice, we conduct ablations on four aspects: (1) the joint diffusion model, (2) the number of sampled joint hypotheses, (3) the source of joint conditions, and (4) the representation used for mesh diffusion model.

Performance of the joint diffusion model. We assess the joint diffusion model by measuring P-MPJPE on its generated joints, using both the averaged and the best hypothesis (Tab. 4). The model is competitive in direct pose estimation on FreiHAND, while errors are higher on the hand-object datasets HO3Dv2 and DexYCB, where object-induced occlusion and complex interaction make lifting 2D keypoints to 3D harder. Across the datasets, the averaged error stays nearly constant as the number of samples grows, whereas the best hypothesis steadily improves. This indicates that the samples are spread around the target rather than collapsing onto a single prediction. Since the quality of these joints

directly affects the final mesh, this underlines the importance of reliable joint predictions in our cascaded framework.

Number of sampling. We analyze how the number of sampled joint hypotheses N affects performance. As shown in Tab. 4, increasing N consistently improves the best-case P-MPJPE, while the averaged performance stays nearly unchanged. Larger N therefore does not improve a single prediction better, instead expands the set of hypotheses and raises the chance that a high-quality one is present. We also observe that, for a fixed joint condition, the Mesh LDM itself produces little variance, since it primarily refines the given coarse joint into a mesh. These two observations together clarify the role of diversity in our framework: the diversity comes from the joint stage and is exploited as a signal during training, while the mesh stage acts as a stable refiner rather than a source of output variance. This is consistent with our inference scheme, where sampled joints and latents are aggregated into a stable estimate.

Joint condition	P-MPJPE ↓	P-MPVPE ↓	F@5 ↑	F@15 ↑
off-the-shelf [52]	5.1	5.4	0.800	0.991
Averaged joint	5.1	5.4	0.805	0.992
Proposed	5.0	5.2	0.816	0.992
Proposed (best)	4.2	4.5	0.866	0.995

Table 5: Ablation study of different joint condition sources on FreiHAND. Conditioning the Mesh LDM on diverse joint hypotheses during training outperforms single-conditioned variants, improving accuracy and generalization across joint conditions. "best" denotes an oracle best-of-32 selection.

Source of joint conditions. This ablation isolates the central claim of our framework: that exposing the mesh stage to *diverse* joint conditions during training, rather than a single deterministic one, is what drives the improvement. We compare three sources of joint conditions for the Mesh LDM: (1) a single joint from an off-the-shelf estimator [52], (2) the averaged joint from our joint diffusion model, and (3) diverse joint hypotheses from our full cascaded pipeline. During training, (1) and (2) condition the Mesh LDM on a single joint per image, whereas (3) conditions it on a joint sampled from the diffusion distribution, so it varies across iterations. At inference, all three condition on a single joint (the estimator output for (1), and the averaged joint for (2) and (3)), so the inference-time input is comparable across settings.

As shown in Tab. 5, the two single-condition variants (1) and (2) perform almost identically (5.1 P-MPJPE), even though one uses an external estimator and the other uses our own averaged joint. What separates them from our full model is not the *source* of the joint, but whether the mesh stage saw a *distribution* of joints during training: only the diverse-conditioned model (3) improves (5.0 P-MPJPE and consistently higher F-scores). This isolates diversity in the training condition as the decisive factor. We attribute this to the mesh stage learning

a joint-mesh mapping over a range of plausible joints instead of memorizing a single joint-to-mesh correspondence. As a result, the mesh stage generalizes better across the joint estimates it encounters at test time. In other words, accurate mesh reconstruction depends not only on the mesh model itself, but also on the distributional quality of the joints it is trained on. This is the purpose of our cascaded design. The two-stage structure is what makes diverse conditioning possible, and it is this diversity that drives the gain.

Diffusion space	FreiHAND		HO3Dv2	
	P-MPJPE ↓	P-MPVPE ↓	P-MPJPE ↓	P-MPVPE ↓
MANO	5.70	5.82	7.81	7.81
Ours (latent)	5.00	5.23	7.50	7.52

Table 6: Quantitative comparison of diffusion in MANO space and our learned latent space. We evaluate both representations using the same cascaded diffusion framework on the FreiHAND and HO3Dv2 datasets.

Representation for mesh diffusion model. Finally, we compare our learned latent representation with MANO [55] parameters as the target space for diffusion. As shown in Tab. 6, diffusion in our latent space achieves lower errors on both benchmarks. Although MANO provides a strong shape prior, every reconstruction is confined to its parameter manifold, which limits fine-grained articulation. Our latent space is free of this constraint and preserves both surface geometry and articulation. We therefore diffuse in this learned latent rather than in raw 3D vertex space, which tends to lose surface and pose structure (Fig. 2(a)). Overall, a learned latent space is an effective target for diffusion-based mesh reconstruction in our pipeline.

5 Conclusions

We presented a coarse-to-fine cascaded diffusion framework for 3D hand pose estimation, combining a joint diffusion model and a Mesh Latent Diffusion Model (Mesh LDM). The joint diffusion model generates diverse 3D joint hypotheses, and the Mesh LDM reconstructs a 3D hand mesh conditioned on a joint sample from these hypotheses. Rather than treating the diverse hypotheses as final outputs, we use the hypotheses as a training signal for the mesh stage to learn distribution-aware joint-mesh relationships robust to variations in coarse predictions.

Extensive experiments on the FreiHAND, HO3Dv2, and DexYCB benchmarks show that our method achieves strong performance, reaching the best results on DexYCB, and remains stable under occlusion and pose ambiguity.

For future work, we plan to extend our framework to two-hand interaction and to explicit hand–object modeling, jointly reconstructing the hand and the manipulated object to better handle complex real-world scenarios.

Acknowledgements

This work was supported by NST grant (CRC 21015, CRC23023-000, MSIT), IITP grant (RS-2023-00228996, RS-2024- 00459749, RS-2025-25443318, RS-2025-25441313, RS-2026-25526850, RS-2026-25522885, MSIT) , KOCCA grant (RS-2024-00442308, MCST) and InnoCORE program (N10260110, MSIT)

References

1. Armagan, A., Garcia-Hernando, G., Baek, S., Hampali, S., Rad, M., Zhang, Z., Xie, S., Chen, M., Zhang, B., Xiong, F., et al.: Measuring generalisation to unseen viewpoints, articulations, shapes and objects for 3d hand pose estimation under hand-object interaction. In: ECCV (2020)
2. Baek, S., Kim, K.I., Kim, T.K.: Pushing the envelope for rgb-based dense 3d hand pose estimation via neural rendering. In: CVPR (2019)
3. Baek, S., Kim, K.I., Kim, T.K.: Weakly-supervised domain adaptation via gan and mesh model for estimating 3d hand poses interacting objects. In: CVPR (2020)
4. Boukhayma, A., Bem, R.d., Torr, P.H.: 3d hand shape and pose from images in the wild. In: CVPR (2019)
5. Chao, Y.W., Yang, W., Xiang, Y., Molchanov, P., Handa, A., Tremblay, J., Narang, Y.S., Van Wyk, K., Iqbal, U., Birchfield, S., et al.: Dexycb: A benchmark for capturing hand grasping of objects. In: CVPR (2021)
6. Chen, P., Chen, Y., Yang, D., Wu, F., Li, Q., Xia, Q., Tan, Y.: I2uv-handnet: Image-to-uv prediction network for accurate and high-fidelity 3d hand mesh modeling. In: ICCV (2021)
7. Chen, X., Liu, Y., Dong, Y., Zhang, X., Ma, C., Xiong, Y., Zhang, Y., Guo, X.: Mobrecon: Mobile-friendly hand mesh reconstruction from monocular image. In: CVPR (2022)
8. Chen, X., Song, Z., Jiang, X., Hu, Y., Yu, J., Zhang, L.: Handos: 3d hand reconstruction in one stage. In: CVPR (2025)
9. Cheng, W., Tang, H., Van Gool, L., Ko, J.H.: Handdiff: 3d hand pose estimation with diffusion on image-point cloud. In: CVPR (2024)
10. Cho, H., Kim, J.: Generative approach for probabilistic human mesh recovery using diffusion models. In: ICCVW (2023)
11. Cho, W., Lee, J., Yi, M., Kim, M., Woo, T., Kim, D., Ha, T., Lee, H., Ryu, J.H., Woo, W., et al.: Dense hand-object (ho) graspnet with full grasping taxonomy and dynamics. In: ECCV (2024)
12. Choi, H., Moon, G., Lee, K.M.: Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In: ECCV (2020)
13. Dong, H., Chharia, A., Gou, W., Carrasco, F.V., De la Torre, F.: Hamba: Single-view 3d hand reconstruction with graph-guided bi-scanning mamba. NeurIPS (2024)
14. Feng, R., Gao, Y., Tse, T.H.E., Ma, X., Chang, H.J.: Diffpose: Spatiotemporal diffusion model for video-based human pose estimation. In: ICCV (2023)
15. Foo, L.G., Gong, J., Rahmani, H., Liu, J.: Distribution-aligned diffusion for human mesh recovery. In: ICCV (2023)
16. Ge, L., Ren, Z., Li, Y., Xue, Z., Wang, Y., Cai, J., Yuan, J.: 3d hand shape and pose estimation from a single rgb image. In: CVPR (2019)

17. Gong, J., Foo, L.G., Fan, Z., Ke, Q., Rahmani, H., Liu, J.: Diffpose: Toward more reliable 3d pose estimation. In: CVPR (2023)
18. Gong, S., Chen, L., Bronstein, M., Zafeiriou, S.: Spiralnet++: A fast and highly efficient mesh convolution operator. In: ICCV (2019)
19. Guleryuz, O.G., Kaeser-Chen, C.: Fast lifting for 3d hand pose estimation in ar/vr applications. In: ICIP (2018)
20. Hampali, S., Rad, M., Oberweger, M., Lepetit, V.: Honnotate: A method for 3d annotation of hand and object poses. In: CVPR (2020)
21. Hampali, S., Sarkar, S.D., Rad, M., Lepetit, V.: Keypoint transformer: Solving joint identification in challenging hands and object interactions for accurate 3d pose estimation. In: CVPR (2022)
22. Handa, A., Van Wyk, K., Yang, W., Liang, J., Chao, Y.W., Wan, Q., Birchfield, S., Ratliff, N., Fox, D.: Dexpilot: Vision-based teleoperation of dexterous robotic hand-arm system (2020)
23. Hasson, Y., Varol, G., Tzionas, D., Kalevatykh, I., Black, M.J., Laptev, I., Schmid, C.: Learning joint reconstruction of hands and manipulated objects. In: CVPR (2019)
24. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
25. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. NeurIPS (2020)
26. Ho, J., Saharia, C., Chan, W., Fleet, D.J., Norouzi, M., Salimans, T.: Cascaded diffusion models for high fidelity image generation. JMLR (2022)
27. Holmquist, K., Wandt, B.: Diffpose: Multi-hypothesis human pose estimation using diffusion models. In: ICCV (2023)
28. Ivashechkin, M., Mendez, O., Bowden, R.: Denoising diffusion for 3d hand pose estimation from images. In: ICCV (2023)
29. Jain, S., Watson, D., Tabellion, E., Poole, B., Kontkanen, J., et al.: Video interpolation with diffusion models. In: CVPR (2024)
30. Jiang, Z., Rahmani, H., Black, S., Williams, B.M.: A probabilistic attention model with occlusion-aware texture regression for 3d hand reconstruction from a single rgb image. In: CVPR (2023)
31. Koo, J., Yoo, S., Nguyen, M.H., Sung, M.: Salad: Part-level latent diffusion for 3d shape generation and manipulation. In: ICCV (2023)
32. Kulon, D., Guler, R.A., Kokkinos, I., Bronstein, M.M., Zafeiriou, S.: Weakly-supervised mesh-convolutional hand reconstruction in the wild. In: CVPR (2020)
33. Lee, J., Jang, J., Kim, D., Sung, M., Kim, T.K.: Fourierhandflow: Neural 4d hand representation using fourier query flow. NeurIPS (2023)
34. Lee, J., Saito, S., Nam, G., Sung, M., Kim, T.K.: Interhandgen: Two-hand interaction generation via cascaded reverse diffusion. In: CVPR (2024)
35. Lee, J., Sung, M., Choi, H., Kim, T.K.: Im2hands: Learning attentive implicit representation of interacting two-hand shapes. In: CVPR (2023)
36. Lee, J., Xu, W., Richard, A., Wei, S.E., Saito, S., Bai, S., Wang, T.L., Sung, M., Kim, T.K., Saragih, J.: Rewind: Real-time egocentric whole-body motion diffusion with exemplar-based identity conditioning. In: CVPR (2025)
37. Li, H., Basri, R., Kluger, Y.: Likelihood training of cascaded diffusion models via hierarchical volume-preserving maps. ICLR (2024)
38. Li, M., Zhang, H., Zhang, Y., Shao, R., Yu, T., Liu, Y.: Hhmr: Holistic hand mesh recovery by enhancing the multimodal controllability of graph diffusion models. In: CVPR (2024)

39. Li, S., Ma, X., Liang, H., Görner, M., Ruppel, P., Fang, B., Sun, F., Zhang, J.: Vision-based teleoperation of shadow dexterous hand using end-to-end deep neural network (2019)
40. Liang, H., Zhang, W., Li, W., Yu, J., Xu, L.: Intergen: Diffusion-based multi-human motion generation under complex interactions. IJCV (2024)
41. Lin, K., Wang, L., Liu, Z.: End-to-end human pose and mesh reconstruction with transformers. In: CVPR (2021)
42. Lin, K., Wang, L., Liu, Z.: Mesh graphormer. In: ICCV (2021)
43. Liu, S., Jiang, H., Xu, J., Liu, S., Wang, X.: Semi-supervised 3d hand-object poses estimation with interactions in time. In: CVPR (2021)
44. Lopez, P.R., Oh, J.H., Jeong, J.G., Jung, H., Lee, J.H., Jaramillo, I.E., Chola, C., Lee, W.H., Kim, T.S.: Dexterous object manipulation with an anthropomorphic robot hand via natural hand pose transformer and deep reinforcement learning. Applied Sciences (2023)
45. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
46. Lu, J., Lin, J., Dou, H., Zeng, A., Deng, Y., Zhang, Y., Wang, H.: Dposer: Diffusion model as robust 3d human pose prior. arxiv:2312.05541 (2023)
47. Moon, G., Lee, K.M.: I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In: ECCV (2020)
48. Park, J., Oh, Y., Moon, G., Choi, H., Lee, K.M.: Handoccnet: Occlusion-robust 3d hand mesh estimation network. In: CVPR (2022)
49. Pavlakos, G., Shan, D., Radosavovic, I., Kanazawa, A., Fouhey, D., Malik, J.: Reconstructing hands in 3d with transformers. In: CVPR (2024)
50. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: ICCV (2023)
51. Perez, E., Strub, F., De Vries, H., Dumoulin, V., Courville, A.: Film: Visual reasoning with a general conditioning layer. In: AAAI (2018)
52. Potamias, R.A., Zhang, J., Deng, J., Zafeiriou, S.: Wilor: End-to-end 3d hand localization and reconstruction in-the-wild. In: CVPR (2025)
53. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation (2021)
54. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022)
55. Romero, J., Tzionas, D., Black, M.J.: Embodied hands: modeling and capturing hands and bodies together. ACM Transactions on Graphics (TOG) (2017)
56. Salimans, T., Ho, J.: Progressive distillation for fast sampling of diffusion models. ICLR (2022)
57. Shabani, M.A., Wang, Z., Liu, D., Zhao, N., Yang, J., Furukawa, Y.: Visual layout composer: Image-vector dual diffusion model for design layout generation. In: CVPR (2024)
58. Shafir, Y., Tevet, G., Kapon, R., Bermano, A.H.: Human motion diffusion as a generative prior. ICLR (2023)
59. Shan, W., Liu, Z., Zhang, X., Wang, Z., Han, K., Wang, S., Ma, S., Gao, W.: Diffusion-based 3d human pose estimation with multi-hypothesis aggregation. In: ICCV (2023)
60. Sharma, A., Hedderich, M.A., Bhardwaj, D., Fruchard, B., McIntosh, J., Nittala, A.S., Klakow, D., Ashbrook, D., Steimle, J.: Solofinger: Robust microgestures while grasping everyday objects. In: Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (2021)

61. Sharma, A., Roo, J.S., Steimle, J.: Grasping microgestures: Eliciting single-hand microgestures for handheld objects. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (2019)
62. Shi, Y., Zhao, L., Lu, X., Hoang, T., Wang, M.: Grasping 3d objects with virtual hand in vr environment. In: SIGGRAPH (2022)
63. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. ICLR (2020)
64. Spurr, A., Iqbal, U., Molchanov, P., Hilliges, O., Kautz, J.: Weakly supervised 3d hand pose estimation via biomechanical constraints. In: ECCV (2020)
65. Tang, X., Wang, T., Fu, C.W.: Towards accurate alignment in real-time 3d hand-mesh reconstruction. In: ICCV (2021)
66. Tevet, G., Raab, S., Gordon, B., Shafir, Y., Cohen-or, D., Bermano, A.H.: Human motion diffusion model. In: ICLR (2023)
67. Woo, T., Kim, T.K., Park, J.: Hand-object reconstruction via interaction-aware graph attention mechanism. In: ICIP. IEEE (2024)
68. Xu, H., Wang, T., Tang, X., Fu, C.W.: H2onet: Hand-occlusion-and-orientation-aware network for real-time 3d hand mesh reconstruction. In: CVPR (2023)
69. Xu, Y., Zhang, J., Zhang, Q., Tao, D.: Vitpose: Simple vision transformer baselines for human pose estimation. NeurIPS (2022)
70. Yang, L., Li, K., Zhan, X., Lv, J., Xu, W., Li, J., Lu, C.: Artiboost: Boosting articulated 3d hand-object pose estimation via online exploration and synthesis. In: CVPR (2022)
71. Zhang, J., Tu, Z., Yang, J., Chen, Y., Yuan, J.: Mixste: Seq2seq mixed spatio-temporal encoder for 3d human pose estimation in video. In: CVPR (2022)
72. Zhang, S., Wang, J., Zhang, Y., Zhao, K., Yuan, H., Qin, Z., Wang, X., Zhao, D., Zhou, J.: I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models. arXiv preprint arXiv:2311.04145 (2023)
73. Zhang, X., Li, Q., Mo, H., Zhang, W., Zheng, W.: End-to-end hand mesh recovery from a monocular rgb image. In: ICCV (2019)
74. Zhou, Z., Zhou, S., Lv, Z., Zou, M., Tang, Y., Liang, J.: A simple baseline for efficient hand mesh reconstruction. In: CVPR (2024)
75. Zimmermann, C., Ceylan, D., Yang, J., Russell, B., Argus, M., Brox, T.: Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In: ICCV (2019)

Cascaded Diffusion Framework for Probabilistic Coarse-to-Fine Hand Pose Estimation

Supplementary Material

In this supplementary material, we provide implementation details of our cascaded diffusion model. We also present additional results on the FreiHAND [75] and HO3Dv2 [20] datasets.

A Implementation details

A.1 Joint diffusion model

Our joint diffusion model is adapted from the D3DP framework [59], originally designed for lifting 2D human pose sequences to 3D using the MixSTE backbone [71]. However, since our goal is estimating single-frame 3D hand poses, we modify the sequence length to 1. The model employs a hidden dimension of 512 and includes 8 MixSTE blocks. We normalize 3D hand joints during training.

The model is trained using only the diffusion loss $\mathcal{L}_{\text{DDPM}}$, with a linear noise scheduler ($\beta \in [0.0001, 0.01]$) with 1000 diffusion timesteps. Input 2D keypoints are obtained from an off-the-shelf estimator [52]. To enhance generalization, we apply data augmentation by randomly rotating the 2D keypoints and 3D hand joints within [-60°, 60°] and scaling the 2D keypoints within [0.9, 1.1]. The joint diffusion model is trained for 250K steps with an initial learning rate of 1e-4, which decays by a factor of 0.9 every 20K steps using a step-based learning rate schedule.

A.2 Mesh AutoEncoder

The Mesh AutoEncoder (Mesh AE) is based on SpiralNet++ framework [18], which encodes 3D mesh using spiral convolutions. Given a hand mesh with vertex positions $V \in \mathbb{R}^{778\times 3}$, the encoder $\mathcal{E}$ compresses the mesh into a latent representation $x \in \mathbb{R}^{168}$, while the decoder $\mathcal{D}$ reconstructs the hand mesh from the latent vector:

$$x = \mathcal{E}(V), \hat{V} = \mathcal{D}(x). \tag{3}$$

Note that the hand mesh V is mean-centered, *i.e.* $\bar{V} = \mathbf{0}$. We employ the following loss terms during training:

- **Vertex Loss** $\mathcal{L}_V$: L1 loss between the ground-truth mesh vertices V and the reconstructed vertices $\hat{V}$, encouraging accurate mesh reconstruction.
- **Joint Loss** $\mathcal{L}_J$: L1 loss between the ground-truth hand joint J and the reconstructed joint $\hat{J} = \mathcal{J}\hat{V}$. $\mathcal{J}$ is a joint regressor matrix.

- **KL Regularization** $\mathcal{L}_{KL}$: We apply KL divergence to the latent vector of AE to follow a Gaussian distribution, which improves the generalization for Mesh LDM.
- **Loss configuration**:

$$\mathcal{L} = \lambda_V \mathcal{L}_V + \lambda_J \mathcal{L}_J + \lambda_{KL} \mathcal{L}_{KL} \tag{4}$$

where $\lambda_V = 1, \lambda_J = 0.5, \lambda_{KL} = 1e-3$.

Mesh AE is trained for 1000 epochs with batch size of 50 with AdamW optimizer [45]. The initial learning rate is 1e-3 and decays a factor of 0.9 every 50 epochs. Note that for HO3Dv2 [20] training, the initial learning rate is 1e-4.

A.3 Mesh LDM

Mesh LDM reconstructs the latent vector of a 3D hand mesh by denoising a noisy latent vector: $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$, where $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i, \epsilon \sim N(0, I)$, and α_t is noise variance schedule at timestep t. It is conditioned on: (1) the reconstructed 3D joint $\hat{J}_0$ from the joint diffusion model and (2) image features $\mathcal{I}$ extracted from an image encoder. The image features $\mathcal{I}$ consist of four levels of extracted features: $\mathcal{I} = \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4\}$.

During training on FreiHAND [75], we use ViT-based features from an off-the-shelf encoder [52]. For HO3Dv2 [20], we use ResNet-50 [24], where each stage's output is treated as a feature level. In the ViT-based case, a single global image feature is upsampled using three deconvolution layers.

Architecture. Mesh LDM follows the DiT framework [50], employing a transformer-based architecture. The input latent vector is repeated 21 times, and concatenated with the denoised joint $\hat{J}_0$. The resulting input tensor has a shape of $\mathbb{R}^{171 \times 21}$, where 171 is the channel dimension, and 21 is the sequence length. The input tensor is tokenized, and the channel dimension is expanded to 512. The Mesh LDM consists of 12 transformer blocks with 16 attention heads. Each Mesh LDM block processes the input as follows:

(1) **Cross-attention**: conducting cross-attention with each image feature and concatenate the results.
(2) **Self-attention and MLP**: Similar to DiT, these layers refine the latent representation.
(3) **Output layer**: The denoised latent $\hat{x}_0$ outputs through reshape function.

We apply Adaptive layer normalization [51] between layers to each level of layers. Finally, the output dimension is 8, and the final tensor is flatten the final tensor to reconstruct 168-dimensional latent vector.

Training. The Mesh LDM is trained for 100K steps with a learning rate of 1e-4, using 1000 diffusion timesteps, and decays by a factor of 0.9 every 5K steps. The training loss includes:

- **Diffusion loss** $\mathcal{L}_{DDPM}$: L2 loss between the ground-truth latent vector x_0 and the reconstructed latent $\hat{x}_0$.

- **Vertex Loss** $\mathcal{L}_V$: L1 loss between the ground-truth mesh vertices V and the reconstructed vertices $\hat{V}$, encouraging accurate mesh reconstruction.
- **Joint Loss** $\mathcal{L}_J$: L1 loss between the ground-truth hand joint J and the reconstructed joint $\hat{J} = \mathcal{J}\hat{V}$. $\mathcal{J}$ is a joint regressor matrix.
- **Loss configuration**:

$$\mathcal{L} = \lambda_{DDPM}\mathcal{L}_{DDPM} + \lambda_V\mathcal{L}_V + \lambda_J\mathcal{L}_J, \tag{5}$$

  where $\lambda_{DDPM} = 1, \lambda_V = 10, \lambda_J = 5$.

We also apply rotation augmentations to images and corresponding 3D hand joints and hand mesh.

A.4 Details for MANO Mesh LDM

For the ablation study, we also evaluate a variant of Mesh LDM that predicts latent vector in the MANO parameter [55] space. In implementation, the 58-dimensional MANO parameters are repeated 21 times, similar to original models. Then, the repeated vectors and the 3D joint coordinates (21×3) are concatenated, forming a 61-dimensional latent vector with 21 sequence lengths ($\mathbb{R}^{61\times21}$). As the output vector's shape is $\mathbb{R}^{168}$, we change the shape of it with MLP layer to form $\mathbb{R}^{58}$ vectors.

B Additional Results

Image encoders	P-MPJPE ↓	P-MPVPE ↓	F@5 ↑	F@15 ↑
off-the-shelf	5.00	5.23	0.816	0.992
ViTPose-B	5.02	5.26	0.811	0.992

Table 7: Comparison of mesh reconstruction performance using different image encoders. Both the off-the-shelf encoder and ViTPose-B yield comparable results across all metrics, demonstrating the robustness of our cascaded framework to encoder variation on FreiHAND dataset.

B.1 Variant of image encoder

To assess the effect of the image encoder, we additionally train our model using a ViTPose-B encoder [69]. Table 7 reports the comparison results under the same evaluation protocol as the main experiments. Although ViTPose-B is trained on a smaller dataset than the off-the-shelf encoder, the performance differences are comparable, indicating that our cascaded framework is robust to variations in encoder architecture and generalizes well across different image encoders.

Number of samples	P-MPJPE ↓	P-MPVPE ↓	F@5 ↑	F@15 ↑
1	5.0	5.2	0.816	0.992
5	4.7	4.7	0.835	0.994
10	4.6	4.6	0.844	0.994
50	4.2	4.5	0.866	0.995

Table 8: Effect of the number of samples from the cascaded diffusion model. The table reports metrics of the *final* hand mesh on FreiHAND. In contrast to Tab. 4, which evaluates the Stage-1 joint output, this measures the full cascaded pipeline.

B.2 Multi-hypotheses

For the FreiHAND dataset, we also analyze the impact of the number of samples on the performance of the cascaded diffusion model. Specifically, we generate 50 joint hypotheses from the joint diffusion model and feed them into the Mesh LDM. The corresponding quantitative results are presented in Table 8. Similar to the joint diffusion model, as the number of generated samples increases, the best performance of cascaded diffusion model also improves.

While multi-sampling improves quantitative performance, the visual differences between generated samples are subtle. Notably, the variations among hypotheses primarily affect the hand's shape rather than its pose. This occurs because the joint diffusion model generates a joint hypothesis, which then conditions Mesh LDM. At this stage, the pose configuration is already determined, and Mesh LDM reconstructs the hand mesh based on the given joint sample. Additionally, the reconstructed mesh aligns with the input image while refining shape properties such as finger thickness.

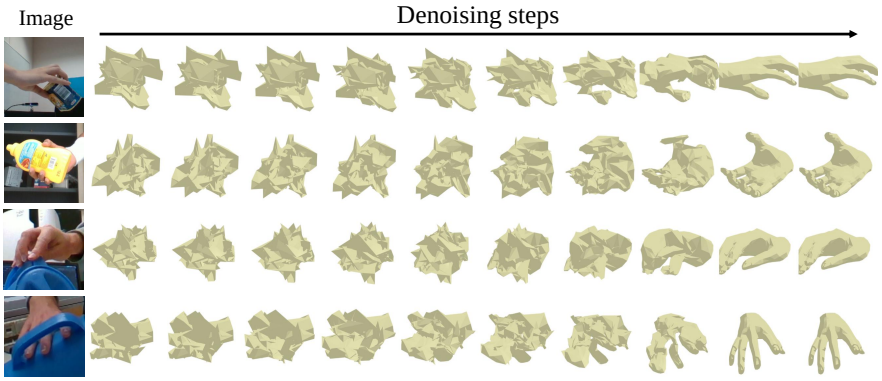

Fig. 7: DDIM denoising process on HO3Dv2 with DDIM step 10.

B.3 DDIM step

Figure 7 visualizes the denoising process of a hand mesh during DDIM inference. Initially, the hand mesh exhibits minimal structural definition. As the DDIM process progresses, the hand pose becomes more articulated after half of the DDIM steps. After that surface details of hand mesh gradually emerge. This demonstrates that diffusion in the latent space effectively captures both pose and surface information throughout the denoising process.

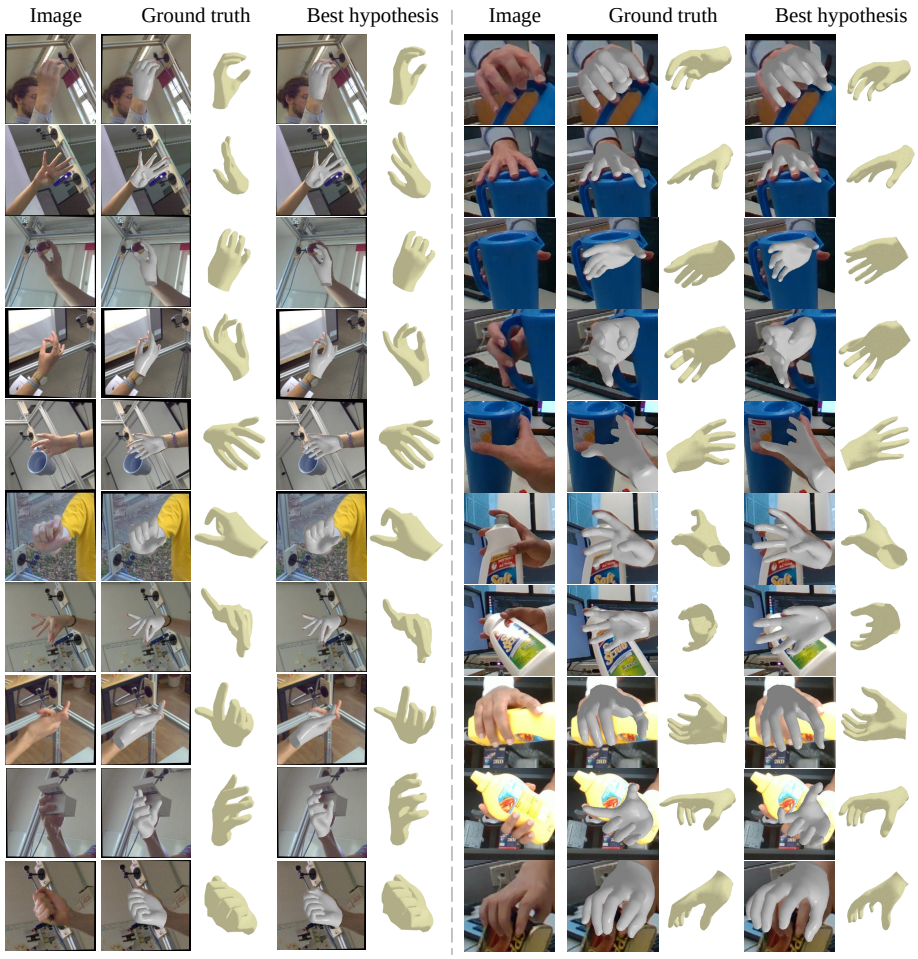

Fig. 8: Qualitative results on FreiHAND and HO3Dv2 dataset.

DDIM	Ours 1	Ours 5	Ours 10	Hamba	WiLoR
Time (ms)	40	140	260	40	50

Table 9: Comparison of inference time. Inference time (in milliseconds) across different methods and ours different DDIM steps.

B.4 Inference speed

Table 9 compares the inference times of our method (across different DDIM steps) with Hamba [13] and WiLoR [52], using a ResNet encoder. While the 10-step variant is slower, our 1-step variant matches the speed of existing methods. Furthermore, recent work on one-step distillation [56] suggests that faster variants are feasible. Importantly, our model maintains robustness under occlusions, making it suitable for applications such as robotic grasping and hand–object interaction.