arXiv:2603.26068v2 [cs.CV] 30 Mar 2026

PAD-Hand: Physics-Aware Diffusion for Hand Motion Recovery

Elkhan Ismayilzada[1] Yufei Zhang[2] Zijun Cui[1]

[1]Michigan State University [2]Independent Researcher

{ismayil1, cuizijun}@msu.edu, yufeizhang96@outlook.com

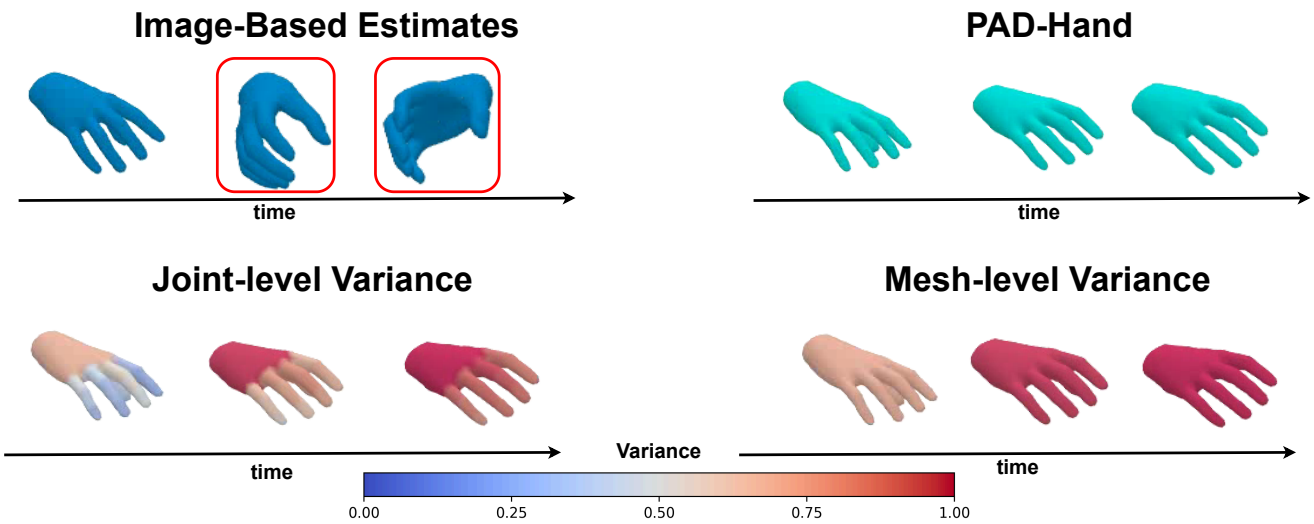

Figure 1. **Refined motion estimates by PAD-Hand with dynamic variance. Top**: Image-based estimates (left) are refined by our model (PAD-Hand) (right) to enforce temporal and physics consistency. **Bottom**: Joint-level (left) and mesh-level (right) variance maps concentrate on frames/regions where the image-based motion estimate is unreliable (highlighted in red), aligning high variance with poor motion estimates. The color bar shows normalized variance (low to high).

Abstract

Significant advancements made in reconstructing hands from images have delivered accurate single-frame estimates, yet they often lack physics consistency and provide no notion of how confidently the motion satisfies physics. In this paper, we propose a novel physics-aware conditional diffusion framework that refines noisy pose sequences into physically plausible hand motion while estimating the physics variance in motion estimates. Building on a MeshCNN–Transformer backbone, we formulate Euler–Lagrange dynamics for articulated hands. Unlike prior works that enforce zero residuals, we treat the resulting dynamic residuals as virtual observables to more effectively integrate physics. Through a last-layer Laplace approximation, our method produces per-joint, per-time variances that measure physics consistency and offers interpretable variance maps indicating where physical consistency weakens. Experiments on two well-known hand datasets show consistent gains over strong image-based initializations and competitive video-based methods. Qualitative results confirm that our variance estimations are aligned with the physical plausibility of the motion in image-based estimates.

1. Introduction

Recovering hand motions from monocular cameras aims to reconstruct a hand's 3D configuration over time from single-view images. This is essential for applications such as augmented and virtual reality (AR/VR) [9, 15, 44, 70], human–computer interaction [23, 47], and beyond.

Following the advancement of deep learning, substantial progress has been made in recovering hand meshes from monocular images [1, 10, 24, 29, 39, 41, 48, 54, 56, 59, 62, 64, 68, 75] or videos [3, 16, 71, 73]. Particularly, scal-

ing up to build foundation models has brought continued performance gains [14, 43, 45], although these models remain solely image-based due to the cost of acquiring video data. They improve per-frame accuracy, yet temporal inconsistency persists. Moreover, all these approaches primarily capture hand kinematic patterns and remain agnostic to dynamics, which suffers from the physical implausibility and limits their applicability in scenarios such as embodied AI.

To address these challenges, incorporating physics into 3D hand reconstruction represents a promising direction. Some works integrate physics through external simulators in related domains such as full-body motion and hand–object interaction [6, 21, 22, 35, 51, 58]. They obtain initial motion estimates from kinematics-based models and then learn control policies or impose explicit physical constraints to refine the results in a post-hoc manner. Other works instead effectively leverage deep learning and directly build physics-informed models for motion refinement [38, 67]. The most recent work by Zhang *et al.* [69] achieves further improvements by capturing the uncertainty in initial motion estimates using diffusion models. Nonetheless, a key limitation of existing physics-based methods lies in their deterministic incorporation of physics. They enforce physics losses by driving equation residuals to zero rather than modeling a distribution over these residuals. Such approaches assume the observed motion can satisfy physics exactly, with no noise, bias, or model mismatch. However, 3D hand estimates are noisy and physics models are approximate, so forcing zero residuals ignores inherent uncertainty. This perfect-data assumption can lead to a challenging optimization landscape [33] or suboptimal solutions [46]. Deterministic formulations also fail to reveal unreliable cases, limiting interpretability. This motivates a probabilistic formulation of physics incorporation, enabling the model to reason over motion data manifold and shape a distributional solution space.

In this work, we introduce a novel physics-aware diffusion framework that effectively incorporates hand dynamics for 3D hand motion recovery from video. We formulate Euler–Lagrange (EL) dynamics for articulated hands and incorporate it into diffusion models. We treat the resulting dynamic residuals as virtual observables drawn from a distribution. Their likelihood is coupled with the standard visual data terms to guide the reverse diffusion process toward physically admissible trajectories. To obtain interpretable physical consistency, we apply a last-layer Laplace approximation to measure variances over associated dynamics residuals. The estimated variances serve as an indicator of uncertainty in the recovery process, allowing us to identify potentially unreliable segments, as shown in Figure 1. In summary, our contributions are as follows:

- **A probabilistic framework for physics integration.** We cast Euler–Lagrange residuals as *virtual observables* [4, 49] and integrate their likelihood into the diffusion objective, enabling a probabilistic integration of physics.
- **Interpretable physics consistency.** We estimate variances of dynamic residuals through a last-layer Laplace approximation, which provides a principled indicator of recovery uncertainty and helps interpret where physical consistency may weaken.
- **Comprehensive evaluation.** On two widely adopted hand datasets, we improve reconstruction accuracy and physical plausibility. Qualitative results confirm that our variance estimations are aligned with the physical plausibility of the motion in image-based estimates.

2. Related Works

Monocular 3D Hand Reconstruction. Current methods for reconstructing 3D hands from images [2, 5, 8, 26, 43, 65, 66, 68, 72, 74] achieve strong single-frame accuracy and have proven effective in specialized scenarios such as hand–object [1, 10, 18–20, 24, 54, 59] and hand–hand interactions [29, 39, 41, 48, 56, 62, 64, 75]. Recent models [14, 43, 45] trained on larger models and datasets further improve generalization. To further promote temporal consistency, other methods take video input and employ temporal contrastive learning [73] or temporal encoders [16] to capture motion dependency. Due to scarcity of high-quality annotated hand video datasets, these methods may still fail to outperform image-based approaches in per-frame reconstruction accuracy. Instead, some works [3, 38, 67, 69, 71] tackle this limitation by recovering hand motion from image-based estimates. These methods are trained solely on 3D motion capture data without videos, providing strong motion priors that encourage temporal consistency. However, consistency alone is not a sufficient factor for a physically plausible hand motion. Such approaches provide data-driven motion priors ignoring the adherence to the governing laws of motion dynamics.

Physics-Informed Hand Motion Modeling. Existing efforts to incorporate physics largely target hand–object interactions. Some works [12, 60] simulate interactions in virtual environments, using physics-informed deep reinforcement learning to obtain realistic grasping and contact patterns. A common limitation of these approaches is their dependence on physics simulators, which complicates end-to-end learning. Another line of work [6, 21, 22, 51] leverages object information to enforce physical constraints during hand reconstruction, but these methods assume the availability of accurate object geometry and trajectories. They also remain deterministic, offering no mechanism to capture or report predictive uncertainty. More recently, Zhang *et al.* [69] employ a conditional diffusion model regularized by intuitive physics terms to encourage natural hand motion. Although conditioning on motion states derived from hand–object proximity can improve realism near contact,

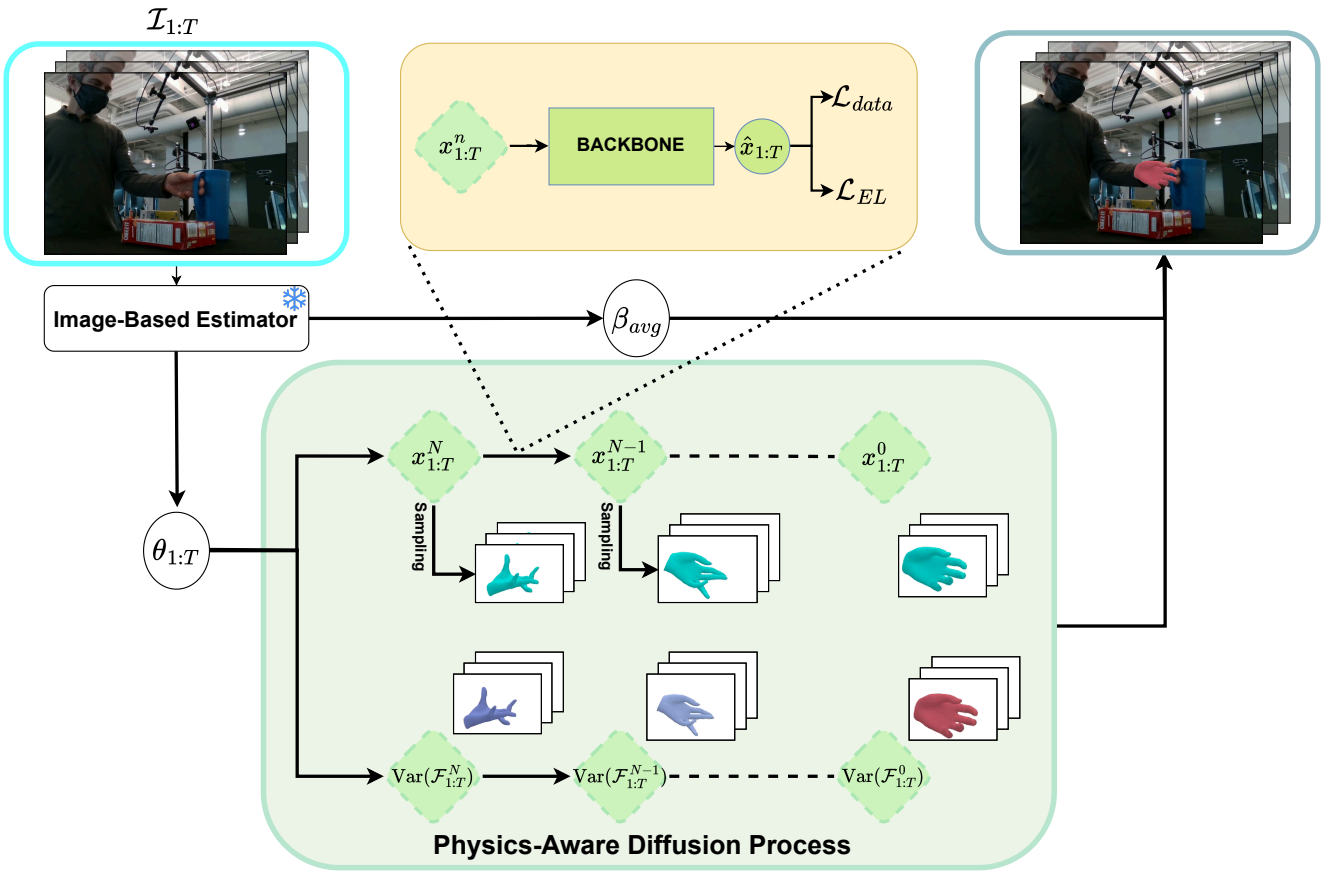

Figure 2. **Overview of PAD-Hand.** A sequence of images $\mathcal{I}_{1:T}$ is passed through an image-based pose estimator to obtain per-frame pose $\theta_{1:T}$ and the average shape β_{avg} estimates. The pose estimates are then refined via a diffusion process to obtain temporally coherent motion. Simultaneously, we propagate the variance at each diffusion step starting from delta Dirac distribution at diffusion step N to obtain per-frame dynamic variance estimates. At each diffusion step, backbone predicts clean motion $\hat{x}_{1:T}$ which is supervised with data-driven loss $\mathcal{L}_{data}$ and physics-driven loss $\mathcal{L}_{EL}$ during training.

this design is inherently object-dependent and provide insufficient information about hand dynamics, such as motion forces. Moreover, existing methods lack a proper mechanism to incorporate physics when the observed motion does not satisfy physics exactly.

Probabilistic Diffusion Model. Several works in the literature propose ways to incorporate the physics into diffusion models [4, 25, 49, 52]. Rixner *et al.* [49] propose to integrate physics residuals as virtual observables and compute ELBO approximation over the joint distribution of the data and the residuals. Bastek *et al.* [4] simplify this process by treating physical likelihood as an auxiliary training objective. However, none of these works propose a way to estimate the variance of their estimations. While several methods quantify uncertainty in image generation [13, 32], such as Kou et al. [32] who apply a last-layer Laplace approximation to obtain pixel-wise uncertainty, we are the first to incorporate physics probabilistically and estimate variance in 3D hand motion recovery.

3. Method

Overview. An overall of our proposed framework is shown in Figure 2. Our pipeline takes a sequence of T frames $\mathcal{I}_{1:T}$ as input and first runs an off-the-shelf hand-pose estimator to obtain per-frame MANO [50] shape $\beta_{1:T}$ and pose $\theta_{1:T}$ estimates. Since hand shape remains constant throughout the sequence, we consider the mean shape $\beta_{avg} = \frac{1}{T}\sum_{i=1}^{T}\beta_i$. Then, we introduce a conditional diffusion model that iteratively refines the motion from a noisy state $x^N_{1:T}$ to a clean state $x^0_{1:T}$ over N diffusion steps, conditioned on the mean shape and per-frame pose estimates $y_{1:T} = \{\beta_{avg}, \theta_{1:T}\}$. Simultaneously, our model produces per-step dynamic variances $\mathrm{Var}(\mathcal{F}^n_{1:T})$ for $n \in [1, N]$. The final output is the refined pose trajectory $x^0_{1:T}$ together with the variance estimates of associated dynamics residuals $\mathrm{Var}(\mathcal{F}^0_{1:T})$. In Section 3.1, we formulate the Euler-Lagrange equation for hand motion, and in Section 3.2, we introduce the conditional diffusion model. In Section 3.3,

we discuss the integration of Euler-Lagrange residual as a virtual observable into the diffusion model, and in section 3.4, we introduce the estimation of the variance of dynamics residuals.

3.1. Euler-Lagrange dynamics for Hand Motion

We build on the MANO parametric model [50]. Its pose parameters $\theta \in \mathbb{R}^{15\times 3}$ define the rotation of 15 hand joints and its shape parameters $\beta \in \mathbb{R}^{10}$ capture the variations across hand shapes. Following established strategies [67], we assume rigid-body dynamics and a constant shape over time, and define the generalized coordinates as $\mathtt{q} = \{R, t, \theta\}$ where $R \in \mathbb{R}^{3\times 3}$ and $t \in \mathbb{R}^3$ are the wrist rotation and translation, respectively. Using this generalized coordinates we can formulate Euler-Lagrange equation as:

$$\mathtt{M}(\mathtt{q};\mathtt{m},\mathtt{I})\ddot{\mathtt{q}} + \mathtt{C}(\mathtt{q},\dot{\mathtt{q}};\mathtt{m},\mathtt{I}) + \mathtt{g}(\mathtt{q};\mathtt{m}) = \mathcal{F} \tag{1}$$

where $\mathtt{M}$, $\mathtt{C}$, $\mathtt{g}$ are the generalized mass matrix, Coriolis, centrifugal terms, gravitational terms, respectively ,and $\mathcal{F}$ on the right-hand side of the equation is the net generalized forces to explain observed hand motion (More explanation on these terms is provided in Appendix A.). Generalized velocity $\dot{\mathtt{q}}$ and acceleration $\ddot{\mathtt{q}}$ can be obtained via the finite difference [36, 57]. To parameterize inertial properties, we obtain a mesh $\mathtt{V}_h \in \mathbb{R}^{778\times 3}$ via linear blend skinning function LBS [50] as $\mathtt{V}_h = \text{LBS}(\theta, \beta)$ and assign each vertex to a kinematic part via MANO skinning weights. For each hand part, we compute volume (via tetrahedralization of the closed surface), mass $\mathtt{m}_k = \rho V_k, k \in \{1, \ldots, 15\}$ under a constant density ρ obtained from literature [40] and local inertia tensor $\mathtt{I}_k$. Refer to [67] for further details.

3.2. Conditional Diffusion Model for Hand Motion Refinement

To recover physically plausible clean motion, we exploit diffusion-based framework, which supports probabilistic inference over trajectories. Specifically, we model a conditional distribution $p(x_{1:T}|y_{1:T})$ for our final refined estimates $x_{1:T}$ conditioned on initial image-based estimates $y_{1:T}$ to approximate data distribution $q(x_{1:T}|y_{1:T})$.

Forward process. Similar to previous works [63, 69], we construct a Markov chain process to gradually shift from clean motion $x_{1:T}$ during training to initial motion estimates $y_{1:T}$ by adding their distance $d_{1:T} = y_{1:T} - x_{1:T}$ through a shifting sequence $\{\eta\}_{n=1}^{N}$ with length N:

$$q(x_{1:T}^n|x_{1:T}^{n-1}, y_{1:T}) = \mathcal{N}(x_{1:T}^n; x_{1:T}^{n-1} + \alpha_n d_{1:T}, \kappa^2\alpha_n I) \tag{2}$$

where κ is the noise variance, $\alpha_n = \eta_n - \eta_{n-1}$ for $n > 1$ and $\alpha_1 = \eta_1$. Given this transition distribution, the marginal distribution at any diffusion step n has a closed-form solution as follows:

$$q(x_{1:T}^n|x_{1:T}, y_{1:T}) = \mathcal{N}(x_{1:T}^n; x_{1:T} + \eta_n d_{1:T}, \kappa^2\eta_n I) \tag{3}$$

To formulate the shifting sequence, we refer to the previous work [63] to monotonically increase η from $\eta_1 \to 0$ to $\eta_N \to 1$. As a result, during the forward process, we start from a Dirac distribution with mean $x_{1:T}$ at diffusion step 0 and end up with a Gaussian distribution $\mathcal{N}(y_{1:T}, \kappa^2 I)$ at diffusion step N.

Reverse process. During the reverse process, we compute the posterior distribution $p(x_{1:T}|y_{1:T})$ as follows:

$$p(x_{1:T}|y_{1:T}) = p(x_{1:T}^N|y_{1:T}) \int \prod_{n=1}^{N} p_\phi(x_{1:T}^{n-1}|x_{1:T}^n, y_{1:T}) dx_{1:T}^{1:N} \tag{4}$$

Here, $p(x_{1:T}^N|y_{1:T})$ is approximately a multivariate Gaussian distribution with mean $y_{1:T}$ and variance κ^2 (i.e., $\mathcal{N}(y_{1:T}, \kappa^2 I)$. The reverse kernel $p_\phi(x_{1:T}^{n-1}|x_{1:T}^n, y_{1:T})$ is the distribution we learn via a backbone with learnable weights ϕ which we introduce in the following subsection.

Backbone. Following the previous work [63], we assume the reverse kernel as a Gaussian distribution, i.e., $p_\phi(x_{1:T}^{n-1}|x_{1:T}^n, y_{1:T}) = \mathcal{N}(x_{1:T}^{n-1}; \mu_\phi(x_{1:T}^n, y_{1:T}, n), \Sigma_n I)$ with fixed variance as $\Sigma_n = \frac{\eta_{n-1}}{\eta_n}\alpha_n\kappa^2$. The target reverse distribution is then formulated as follows:

$$q(x_{1:T}^{n-1}|x_{1:T}^n, x_{1:T}, y_{1:T}) = \mathcal{N}(x_{1:T}^{n-1}|A_n x_{1:T}^n + B_n x_{1:T}, \Sigma_n I) \tag{5}$$

where $A_n = \frac{\eta_{n-1}}{\eta_n}$, $B_n = \frac{\alpha_n}{\eta_n}$. For the derivation of Equation 5 please refer to [63]. To estimate the mean, we reparameterize it as:

$$\mu_\phi(x_{1:T}^n, y_{1:T}, n) = A_n x_{1:T}^n + B_n f_\phi(x_{1:T}^n, y_{1:T}, n) \tag{6}$$

where $f_\phi(x_{1:T}^n, y_{1:T}, n)$ is the backbone with weights ϕ that outputs denoised motion $\hat{x}_{1:T}$. To train the backbone, we optimize following loss function:

$$\mathcal{L}_{data} = \mathbb{E}_{n\sim[1,N]}||x_{1:T} - f_\phi(x_{1:T}^n, y_{1:T}, n)||^2 \tag{7}$$

where $x_{1:T}$ is ground truth motion sampled from data distribution $q(x_{1:T}|y_{1:T})$. The backbone architecture is depicted in Figure 3. A Transformer encoder–decoder is used to model temporal dependencies across frames. We run forward kinematics with the pose and shape parameters (i.e., β_{avg}) to obtain the hand mesh sequence $\mathtt{V}_{1:T}^n$ and extract topology-aware spatial features using a MeshCNN [8]. These features are concatenated with an MLP embedding of the diffusion step n. Up to this point, the model captures geometric and temporal patterns, but remains agnostic to motion dynamics. In the next section, we introduce a mechanism to integrate hand dynamics and render the model physics-aware.

3.3. Physics-aware Diffusion Refinement

Dynamic residual as a virtual observable. Since the output of the backbone is a motion sequence, we can evaluate

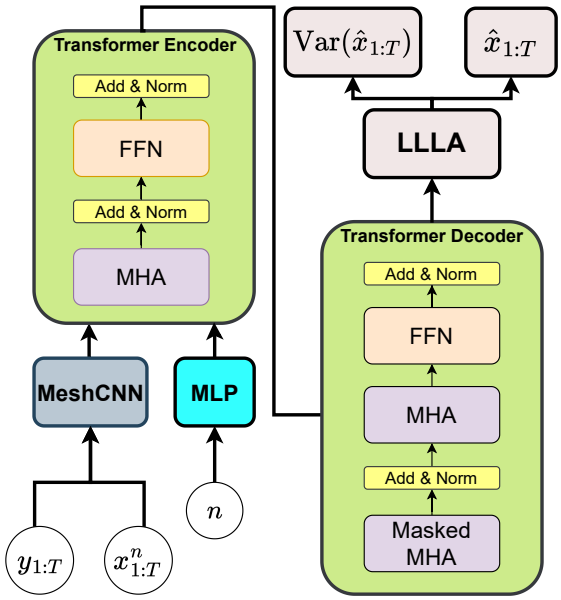

Figure 3. **Backbone architecture**. At diffusion step n, the current pose sequence $x^n_{1:T}$ and image-based estimates $y_{1:T}$ are converted to meshes and encoded by MeshCNN [8], while an MLP encodes n. A Transformer encoder-decoder fuses these features, and an LLLA head predicts the refined pose sequence $\hat{x}_{1:T}$ and its variance $\mathrm{Var}(\hat{x}_{1:T})$.

the Euler-Lagrange residual at each frame as:

$$Z_t = \mathtt{M}_t \ddot{\mathtt{q}}_t + \mathtt{C}_t + \mathtt{g}_t - \hat{\mathcal{F}}_t \tag{8}$$

A direct deterministic penalty would be $\mathcal{L}^D_{EL} = \sum_{t=1}^{T} ||Z_t||_1$ where $\hat{\mathcal{F}}_t$ is the pseudoforce computed from motion data following [67]. Rather than enforcing this loss as a hard constraint, we treat residuals as *virtual observables*. This brings key benefits. The trajectories we refine come from a learned estimator that encodes strong data priors. Forcing them to exactly satisfy an approximate dynamics model (with no explicit contact force modeling, using approximate physical properties such as mass) might push model mismatch into the motion and may distort true hand dynamics. Treating residuals as virtual observations only *scores* how compatible a candidate refinement is with the physics, without overriding the information already present. Therefore, we treat the residuals as virtual observables [4, 49] sampled from the distribution below:

$$q_Z(\zeta_{1:T}|x_{1:T}) = \mathcal{N}(\zeta_{1:T}; Z(x^0_{1:T}), \sigma^2 I) \tag{9}$$

where $x^0_{1:T}$ is the sample from $p(x_{1:T}|y_{1:T})$ by performing a reverse diffusion process during training and σ^2 is a predefined variance. The reason why we choose $x^0_{1:T}$ rather than $\hat{x}_{1:T}$ is because $\hat{x}_{1:T}$ is an estimate of $\mathbb{E}[x^0_{1:T}|x^n_{1:T}]$ and residual evaluation on this estimate is not consistent since $Z(\mathbb{E}[x^0_{1:T}|x^n_{1:T}]) \neq \mathbb{E}[Z(x^0_{1:T}|x^n_{1:T})]$ (Jensen gap [17]). We compute the negative log likelihood loss on the residuals as:

$$\mathcal{L}_{EL} = \mathbb{E}_{n\sim[1,N]} \frac{1}{2\sigma_n} ||Z_{1:T}(x^0_{1:T})||^2 \tag{10}$$

where $\sigma_n = \Sigma_n / c$ is the rescaled variance for the virtual observable.

Total loss. Putting everything together, our framework is trained with the following total objective:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{data} + \lambda_2 \mathcal{L}_{EL} \tag{11}$$

where λ_1, λ_2 are the training weights for $\mathcal{L}_{data}$ and $\mathcal{L}_{EL}$, respectively. During training, we use the motion estimates obtained from an off-the-shelf image-based pose estimator as $y_{1:T}$ to capture the mapping distribution of the pose estimator to motion data.

3.4. Bayesian Inference

To fully capture the conditional distribution, we adopt the Last-layer Laplace Approximation (LLLA) [32] as a post-hoc inference step. Concretely, posterior distribution on the last layer of backbone parameters is mapped to a Gaussian posterior predictive:

$$p(\hat{x}_{1:T}|x^n_{1:T}, n, D) \approx \mathcal{N}(f_\phi(x^n_{1:T}, y_{1:T}, n), \mathrm{diag}(\gamma^2_\phi(x^n_{1:T}, y_{1:T}, n)) \tag{12}$$

where $D = \{(x^s_{1:T}, y^s_{1:T})\}_{s=1}^{S}$ is the training dataset, γ^2_ϕ is the joint-wise variance prediction function. Now we have a variance estimation for the output of the backbone but not for the final output of the diffusion model (i.e., $x^0_{1:T}$). To achieve this, we propagate the uncertainty of the backbone by applying variance estimation on each state. Specifically, sampling for reverse diffusion is formulated as:

$$x^{n-1}_{1:T} = A_n x^n_{1:T} + B_n \hat{x}_{1:T} + \Sigma_n \epsilon \tag{13}$$

where $\epsilon \sim \mathcal{N}(0, I)$. Applying variance estimation to the both sides of equation we get:

$$\mathrm{Var}(x^{n-1}_{1:T}) = A_n^2 \mathrm{Var}(x^n_{1:T}) + B_n^2 \mathrm{Var}(\hat{x}_{1:T}) + \Sigma_n^2 + 2A_n B_n \mathrm{Cov}(x^n_{1:T}, \hat{x}_{1:T}) \tag{14}$$

where $\mathrm{Var}(\hat{x}_{1:T}) = \mathrm{diag}(\gamma^2_\phi(x^n_{1:T}, y_{1:T}, n))$ and

$$\mathrm{Cov}(x^n_{1:T}, \hat{x}_{1:T}) \approx \frac{1}{S}\sum_{i=1}^{S}(x^{n,i}_{1:T} \cdot \hat{x}^i_{1:T}) - \mathbb{E}[x^n_{1:T}] \cdot \frac{1}{S}\sum_{i=1}^{S} \hat{x}^i_{1:T} \tag{15}$$

where $\cdot$ denotes the element-wise multiplication. Expectation of each state is then:

$$\mathbb{E}[x^{n-1}_{1:T}] = A_n \mathbb{E}[x^n_{1:T}] + B_n \mathbb{E}[\hat{x}_{1:T}] \tag{16}$$

Algorithm 1 Variance Estimation for PAD-Hand

Require: Starting point $x^N_{1:T} \sim \mathcal{N}(y_{1:T}, \kappa^2 I)$, Monte Carlo sample size S, Pre-trained backbone f_ϕ.

Ensure: Motion generation $x^0_{1:T}$ and joint-wise uncertainty $\mathrm{Var}(x^0_{1:T})$.

Construct the joint-wise variance prediction function γ^2_ϕ via LLLA (Last Layer Laplace Approximation);
$\mathbb{E}[x^N_{1:T}] \leftarrow x^N_{1:T}$, $\mathrm{Var}(x^N_{1:T}) \leftarrow \mathbf{0}$;
$\mathrm{Cov}(x^N_{1:T}, \hat{x}_{1:T}) \leftarrow \mathbf{0}$;
for $n = N \rightarrow 1$ **do**
  Sample $\hat{x}_{1:T}$ from the Gaussian distribution $\mathcal{N}(f_\phi(x^n_{1:T}, n, y_{1:T}), \mathrm{diag}(\gamma^2_\phi(x^n_{1:T}, n, y_{1:T})))$;
  Obtain $x^{n-1}_{1:T}$ via Equation 13;
  Estimate $\mathbb{E}[x^{n-1}_{1:T}]$ via Equation 16;
  Estimate $\mathrm{Var}(x^{n-1}_{1:T})$ via Equation 14;
  Sample $x^{n-1,\,i}_{1:T} \sim \mathcal{N}(\mathbb{E}[x^{n-1}_{1:T}], \mathrm{Var}(x^{n-1}_{1:T})), i \in \{1, \ldots, S\}$;
  Estimate $\mathrm{Cov}(x^{n-1}_{1:T}, \hat{x}_{1:T})$ via Equation 15;
end for

For the derivations of Equations 14, 15 and 16, please refer to Appendix B. Algorithm 1 demonstrates the procedure to obtain final motion variance $\mathrm{Var}(x^0_{1:T})$. We then compute the dynamics residual variance using the variance of the force inferred from motion since pseudoforce is unavailable at inference. This causes no impact because pseudoforce is only a constant shift and does not change the variance:

$$\mathrm{Var}(\mathcal{F}_{1:T}) \approx J_{\mathcal{F}_{1:T}} \mathrm{Var}(x^0_{1:T}) J^\top_{\mathcal{F}_{1:T}} \tag{17}$$

where $J_{\mathcal{F}_{1:T}} = \frac{\partial \mathcal{F}_{1:T}}{\partial x}\Big|_{x=\mathbb{E}[x^0_{1:T}]}$. $\mathrm{Var}(\mathcal{F}_{1:T})$ measures the uncertainty in the motion recovery process, enabling the identification of potentially unreliable motion estimates.

4. Experiments

4.1. Experimental Settings

Datasets. To evaluate our pipeline, we employ two benchmarks which are DexYCB [7] and HO3D [18]. Both datasets offer high-quality per-frame 3D hand pose and shape annotations, captured with multi-view setups across diverse subjects. We adopt the official training and evaluation splits used in prior studies.

Metrics. Following prior works [16, 43, 45, 72, 73], we report PA-MPJPE (Procrutes aligned Mean Per Joint Error) and MPJPE (Mean Per Joint Error) on DexYCB, and PA-MPJPE (in mm) on HO3D as reconstruction accuracy metrics. For physical plausibility, we additionally report the acceleration error (ACCEL, in mm/frame^2).

Implementation. Our backbone is a Transformer encoder–decoder with four encoder and four decoder layers, each using eight attention heads and an embedding size of 512. For per–frame spatial encoding, the MeshCNN module has four layers with channel dimensions [32, 64, 64, 64]. Since the Transformer in backbone is permutation-invariant, we add standard sinusoidal positional encodings to the frame-wise mesh tokens before feeding them to the Transformer. Following prior works [31, 67, 69], we set the sequence length to $T = 16$. We train with AdamW [30] using an initial learning rate of 2×10^{-4}, decayed by a factor of 0.8 every 10 epochs. The loss weights are $\lambda_1 = 2 \times 10^3$ and $\lambda_2 = 500$. N, S are set to 4 and 20, respectively. In addition, we supervise with geometric losses from [53] and include a consistency regularization term to mitigate error accumulation in diffusion sampling, following [28] (Ablation study on these losses is provided in Appendix C.1).

Table 1. **Comparison to SOTAs on DexYCB**. "*" denotes results obtained from official models; others are from original papers. "P" and "D" denote probabilistic and deterministic models, respectively.

Method	Input	Type	PA-MPJPE↓	MPJPE↓	ACCEL↓
MaskHand [34]	Image	P	5.0	11.70	-
*WiLoR [45]		D	4.88	12.75	6.70
S^{2}HAND(V) [55]	Sequence	D	7.27	19.67	-
VIBE [31]			6.43	16.95	-
TCMR [11]			6.28	16.03	-
*Deformer [16]			5.22	13.64	6.77
BioPR [58]			-	12.81	-
WiLoR + **Ours**	Sequence	P	**4.63**	**10.56**	**3.34**

Table 2. **Comparison to SOTAs on HO3D**. "*" denotes results from official models; others are from original papers. "P" and "D" denote probabilistic and deterministic models, respectively.

Method	Input	Type	PA-MPJPE↓	ACCEL↓
AMVUR [27]	Image	P	10.3	-
*WiLoR [45]		D	7.50	4.98
VIBE [31]	Sequence	D	9.90	-
TCMR [11]			11.40	-
TempCLR [73]			10.60	-
*Deformer [16]			9.40	6.37
WiLoR + **Ours**	Sequence	P	**7.43**	**2.71**

4.2. Quantitative Results

Results on DexYCB. We initialize training with image-based estimates from the large-scale pre-trained WiLoR model [45] and compare our refinements against state-of-the-art video-based hand reconstruction methods. Table 1 reports results on DexYCB. Our framework improves WiLoR on both reconstruction and physical plausibility metrics: PA-MPJPE drops from 4.88 mm to

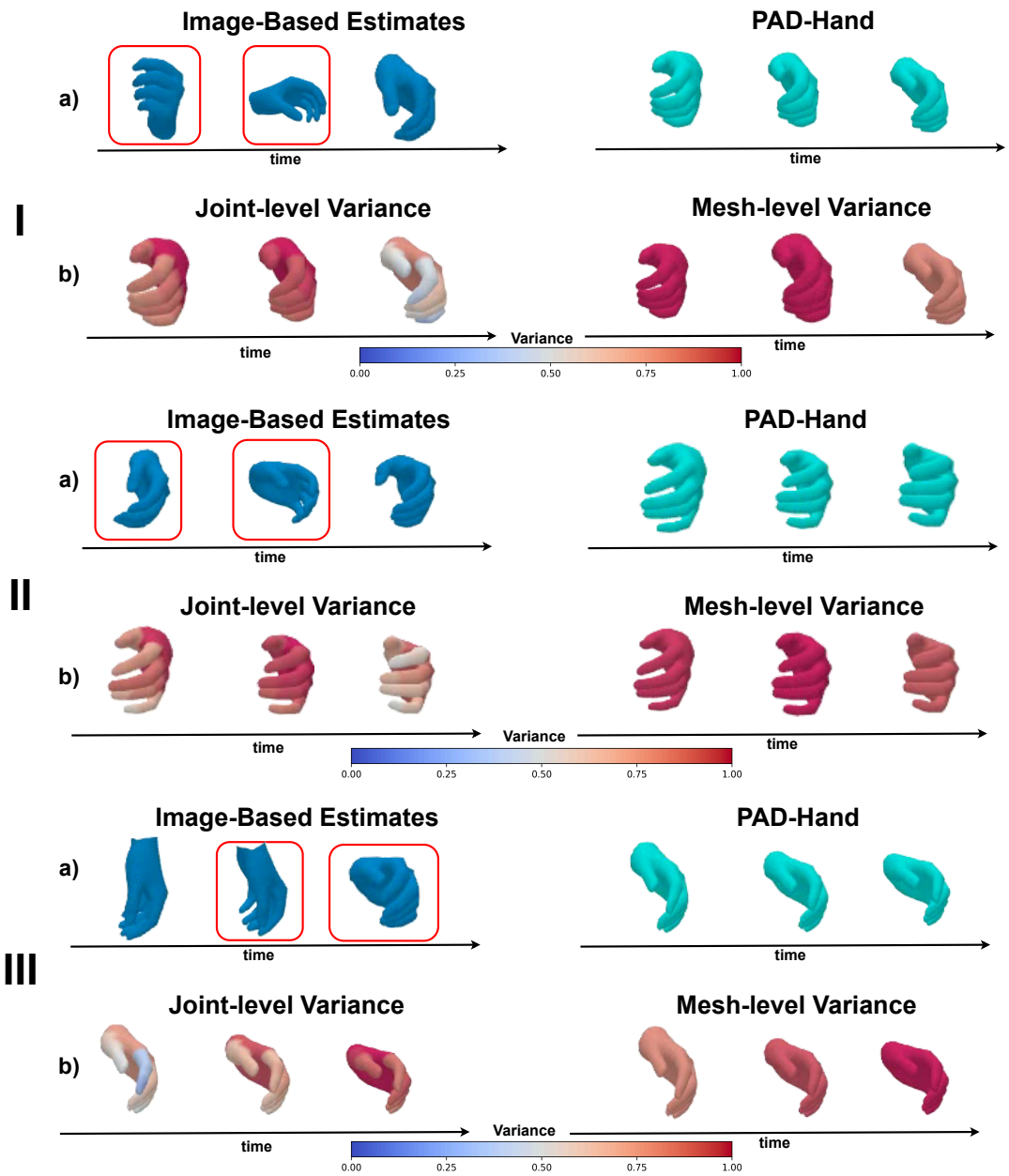

Figure 4. **Refined motion estimates by PAD-Hand with dynamic variance on DexYCB.** We visualize three representative sequences (I–III). In each block, row (a) compares the original image-based motion estimates to the trajectories refined by PAD-Hand, while row (b) shows the corresponding variance estimations in terms of joint-level and mesh-level dynamic variance. The red boxes highlight frames where the image-based estimates exhibit strong jitter.

4.63 mm (5.1%), MPJPE from 12.75 mm to 10.56 mm (17.2%), and the acceleration error from 6.70 mm/frame^2 to 3.34 mm/frame^2 (50.1%). We further report improvements over two additional baselines on DexYCB in Appendix C.2.

Results on HO3D. A similar trend is observed on HO3D. As shown in Table 2, our framework reduces PA-MPJPE from 7.50 mm to 7.43 mm and substantially improves physical consistency, lowering the acceleration error from 4.98 mm/frame^2 to 2.71 mm/frame^2.

4.3. Qualitative Results

Figure 4 qualitatively illustrates the effect of PAD-Hand on three representative sequences from DexYCB. Across all sequences, PAD-Hand produces smoother, more physically plausible motions compared to the jittery image-based estimates. In sequence I (a), a pronounced discontinuity in

global rotation is clearly visible in the image-based motion; our refinement removes this jump and restores a coherent trajectory. The corresponding variance maps in sequence I (b) show elevated values at both joint and mesh level around the same frames, indicating that our force variance reliably flags this problematic segment. The same pattern appears in sequences II and III: PAD-Hand corrects implausible pose changes while the joint-level and mesh-level variances peak exactly where the original image-based estimates are unstable. This alignment supports our claim that PAD-Hand not only improves motion quality but also provides a meaningful indicator of where the underlying estimates are unreliable. HO3D qualitative results are in Appendix D.

4.4. Ablation Study

Effectiveness of physics residual loss. We validate the effectiveness of our physics loss on DexYCB. As shown in Table 3, integrating the Euler-Lagrange residual loss reduces error across all metrics. In terms of reconstruction accuracy, PA-MPJPE decreases from 4.65 mm to 4.63 mm and MPJPE from 10.62 mm to 10.56 mm. For physical plausibility, the acceleration error improves from 3.36 mm/frame^2 to 3.34 mm/frame^2. Improvements are statistically significant as demonstrated in Appendix C.3, and the marginal gains are due to WiLoR's data sufficiency on this in-distribution dataset. As shown in Appendix C.4, more improvements are observed when testing on an out-of-distribution dataset.

Table 3. **Effectiveness of residual loss on DexYCB.**

Loss		Evaluation Metrics		
$\mathcal{L}_{data}$	$\mathcal{L}_{EL}$	PA-MPJPE↓	MPJPE↓	ACCEL↓
*WiLoR [45]		4.88	12.75	6.70
✓		4.65	10.62	3.36
✓	✓	**4.63**	**10.56**	**3.34**

Effectiveness of physics integration. We ablate the role of physics in our probabilistic framework by comparing a deterministic penalty ($\mathcal{L}_{EL}^{D}$) with our probabilistic physics integration. As shown in Table 4, the probabilistic variant consistently performs better, indicating that enforcing physics at the distribution level is preferable and better aligned with the probabilistic nature of diffusion models.

Usefulness of dynamic variance. We visualize the estimated dynamic variances with a histogram whose bars are colored by the mean Euler–Lagrange residual within each bin. As shown in Figure 5, higher variance bins coincide with larger residuals, indicating that increased dynamic variance flags motions that are physically implausible which is typically arising from poor image-based estimates.

Table 4. **Ablation on physics integration on DexYCB**.

Loss	PA-MPJPE↓	MPJPE↓	ACCEL↓
*WiLoR [45]	4.88	12.75	6.70
$\mathcal{L}_{EL}^{D}$	4.66	10.61	3.35
Ours	**4.63**	**10.56**	**3.34**

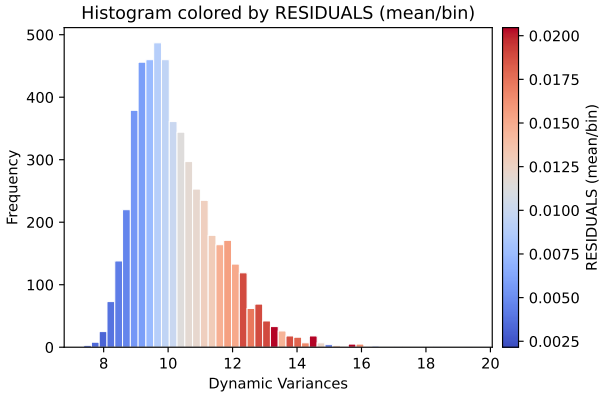

Figure 5. **Distribution of dynamic variances for PAD-Hand.** Bar color encodes the mean Euler–Lagrange residual within each variance bin (blue is low, red is high). Higher variance bins coincide with larger residuals, indicating that the model's uncertainty aligns with physics violations.

5. Conclusion

In this paper, we introduce a physics-aware diffusion framework for 3D hand motion recovery together with its dynamic variance information. We formulate Euler–Lagrange dynamics for hand motion and treat their residuals as virtual observables in a probabilistic model, allowing physics to guide refinement without being enforced as a hard constraint. Experiments on two widely used hand datasets show consistent improvements over both image-based baselines and prior video-based methods. In addition, we propose a method to estimate per-time, per-joint dynamic variances, and demonstrate that these variances reliably highlight frames where the underlying image-based motion estimates are unreliable, providing a physically grounded indicator of motion quality. This work presents an initial use case of dynamic variances, and exploring broader applications of this information remains an interesting direction for future work.

One limitation of our approach is that the current physics modeling does not explicitly capture object geometry or contacts. We consider integrating these aspects for modeling delicate hand–object dynamics as a promising direction for future work.

Acknowledgment

This work was supported by Michigan State University.

References

[1] Ahmed Tawfik Aboukhadra, Jameel Malik, Ahmed Elhayek, Nadia Robertini, and Didier Stricker. Thor-net: End-to-end graformer-based realistic two hands and object reconstruction with self-supervision. In *WACV*, 2023. 1, 2

[2] Seungryul Baek, Kwang In Kim, and Tae-Kyun Kim. Pushing the envelope for rgb-based dense 3d hand pose estimation via neural rendering. In *CVPR*, 2019. 2

[3] Fabien Baradel, Romain Brégier, Thibault Groueix, Philippe Weinzaepfel, Yannis Kalantidis, and Grégory Rogez. Posebert: A generic transformer module for temporal 3d human modeling. *TPAMI*, 2022. 1, 2

[4] Jan-Hendrik Bastek, WaiChing Sun, and Dennis M Kochmann. Physics-informed diffusion models. *ICLR*, 2025. 2, 3, 5

[5] Adnane Boukhayma, Rodrigo Bem, and Philip H.S. Torr. 3d hand shape and pose from images in the wild. In *CVPR*, 2019. 2

[6] Jona Braun, Sammy Christen, Muhammed Kocabas, Emre Aksan, and Otmar Hilliges. Physically plausible full-body hand-object interaction synthesis. In *3DV*, 2024. 2

[7] 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 *CVPR*, 2021. 6

[8] Xingyu Chen, Yufeng Liu, Yajiao Dong, Xiong Zhang, Chongyang Ma, Yanmin Xiong, Yuan Zhang, and Xiaoyan Guo. Mobrecon: Mobile-friendly hand mesh reconstruction from monocular image. In *CVPR*, 2022. 2, 4, 5

[9] Xingyu Chen, Baoyuan Wang, and Heung-Yeung Shum. Hand avatar: Free-pose hand animation and rendering from monocular video. In *CVPR*, 2023. 1

[10] Hoseong Cho, Chanwoo Kim, Jihyeon Kim, Seongyeong Lee, Elkhan Ismayilzada, and Seungryul Baek. Transformer-based unified recognition of two hands manipulating objects. In *CVPR*, 2023. 1, 2

[11] Hongsuk Choi, Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Beyond static features for temporally consistent 3d human pose and shape from a video. In *CVPR*, 2021. 6

[12] Sammy Christen, Muhammed Kocabas, Emre Aksan, Jemin Hwangbo, Jie Song, and Otmar Hilliges. D-grasp: Physically plausible dynamic grasp synthesis for hand-object interactions. In *CVPR*, 2022. 2

[13] Michele De Vita and Vasileios Belagiannis. Diffusion model guided sampling with pixel-wise aleatoric uncertainty estimation. In *WACV*, 2025. 3

[14] Haoye Dong, Aviral Chharia, Wenbo Gou, Francisco Vicente Carrasco, and Fernando D De la Torre. Hamba: Single-view 3d hand reconstruction with graph-guided bi-scanning mamba. *NeurIPS*, 2024. 2

[15] Haoye Dong, Tiange Xiang, Sravan Chittupalli, Jun Liu, and Dong Huang. Physical-space multi-body mesh detection achieved by local alignment and global dense learning. In *WACV*, 2024. 1

[16] Qichen Fu, Xingyu Liu, Ran Xu, Juan Carlos Niebles, and Kris M Kitani. Deformer: Dynamic fusion transformer for robust hand pose estimation. In *ICCV*, 2023. 1, 2, 6

[17] Xiang Gao, Meera Sitharam, and Adrian E Roitberg. Bounds on the jensen gap, and implications for mean-concentrated distributions. *arXiv*, 2017. 5

[18] Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. Honnotate: A method for 3d annotation of hand and object poses. In *CVPR*, 2020. 2, 6

[19] 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 *CVPR*, 2019.

[20] Yana Hasson, Bugra Tekin, Federica Bogo, Ivan Laptev, Marc Pollefeys, and Cordelia Schmid. Leveraging photometric consistency over time for sparsely supervised hand-object reconstruction. In *CVPR*, 2020. 2

[21] Haoyu Hu, Xinyu Yi, Hao Zhang, Jun-Hai Yong, and Feng Xu. Physical interaction: Reconstructing hand-object interactions with physics. In *SIGGRAPH Asia*, 2022. 2

[22] Haoyu Hu, Xinyu Yi, Zhe Cao, Jun-Hai Yong, and Feng Xu. Hand-object interaction controller (hoic): Deep reinforcement learning for reconstructing interactions with physics. In *SIGGRAPH*, 2024. 2

[23] Rongtian Huo, Qing Gao, Jing Qi, and Zhaojie Ju. 3d human pose estimation in video for human-computer/robot interaction. In *ICIRA*, 2023. 1

[24] Elkhan Ismayilzada, MD Khalequzzaman Chowdhury Sayem, Yihalem Yimolal Tiruneh, Mubarrat Tajoar Chowdhury, Muhammadjon Boboev, and Seungryul Baek. Qort-former: Query-optimized real-time transformer for understanding two hands manipulating objects. In *AAAI*, 2025. 1, 2

[25] Christian Jacobsen, Yilin Zhuang, and Karthik Duraisamy. Cocogen: Physically consistent and conditioned score-based generative models for forward and inverse problems. *SIAM Journal on Scientific Computing*, 2025. 3

[26] Changlong Jiang, Yang Xiao, Cunlin Wu, Mingyang Zhang, Jinghong Zheng, Zhiguo Cao, and Joey Tianyi Zhou. A2j-transformer: Anchor-to-joint transformer network for 3d interacting hand pose estimation from a single rgb image. In *CVPR*, 2023. 2

[27] Zheheng Jiang, Hossein Rahmani, Sue Black, and Bryan M Williams. A probabilistic attention model with occlusion-aware texture regression for 3d hand reconstruction from a single rgb image. In *CVPR*, 2023. 6

[28] Dongjun Kim, Chieh-Hsin Lai, Wei-Hsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. Consistency trajectory models: Learning probability flow ode trajectory of diffusion. *arXiv*, 2023. 6

[29] Dong Uk Kim, Kwang In Kim, and Seungryul Baek. End-toend detection and pose estimation of two interacting hands. In *ICCV*, 2021. 1, 2

[30] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2014. 6

[31] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *CVPR*, 2020. 6

[32] Siqi Kou, Lei Gan, Dequan Wang, Chongxuan Li, and Zhijie Deng. Bayesdiff: Estimating pixel-wise uncertainty in diffusion via bayesian inference. *ICLR*, 2023. 3, 5

[33] Aditi Krishnapriyan, Amir Gholami, Shandian Zhe, Robert Kirby, and Michael W Mahoney. Characterizing possible failure modes in physics-informed neural networks. *Advances in neural information processing systems*, 34:26548–26560, 2021. 2

[34] Mengcheng Li, Hongwen Zhang, Yuxiang Zhang, Ruizhi Shao, Tao Yu, and Yebin Liu. Hhmr: Holistic hand mesh recovery by enhancing the multimodal controllability of graph diffusion models. In *CVPR*, 2024. 6

[35] Zongmian Li, Jiri Sedlar, Justin Carpentier, Ivan Laptev, Nicolas Mansard, and Josef Sivic. Estimating 3d motion and forces of person-object interactions from monocular video. In *CVPR*, 2019. 2

[36] C Karen Liu and Sumit Jain. A quick tutorial on multibody dynamics. *Online tutorial, June*, 2012. 4

[37] Yun Liu, Haolin Yang, Xu Si, Ling Liu, Zipeng Li, Yuxiang Zhang, Yebin Liu, and Li Yi. Taco: Benchmarking generalizable bimanual tool-action-object understanding. In *CVPR*, 2024. 2

[38] Haowen Luo, Yunze Liu, and Li Yi. Physics-aware hand-object interaction denoising. In *CVPR*, 2024. 2

[39] Hao Meng, Sheng Jin, Wentao Liu, Chen Qian, Mengxiang Lin, Wanli Ouyang, and Ping Luo. 3d interacting hand pose estimation by hand de-occlusion and removal. In *ECCV*, 2022. 1, 2

[40] Mojtaba Mirakhorlo, Judith MA Visser, BAAX Goislard de Monsabert, FCT Van der Helm, Huub Maas, and HEJ Veeger. Anatomical parameters for musculoskeletal modeling of the hand and wrist. *International Biomechanics*, 2016. 4

[41] Gyeongsik Moon. Bringing inputs to shared domains for 3d interacting hands recovery in the wild. In *CVPR*, 2023. 1, 2

[42] JoonKyu Park, Yeonguk Oh, Gyeongsik Moon, Hongsuk Choi, and Kyoung Mu Lee. Handoccnet: Occlusion-robust 3d hand mesh estimation network. In *CVPR*, 2022. 2

[43] Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3d with transformers. In *CVPR*, 2024. 2, 6

[44] Siyou Pei, Alexander Chen, Jaewook Lee, and Yang Zhang. Hand interfaces: Using hands to imitate objects in ar/vr for expressive interactions. In *CHI*, 2022. 1

[45] Rolandos Alexandros Potamias, Jinglei Zhang, Jiankang Deng, and Stefanos Zafeiriou. Wilor: End-to-end 3d hand localization and reconstruction in-the-wild. *CVPR*, 2025. 2, 6, 8, 1, 3

[46] Ibai Ramirez, Jokin Alcibar, Joel Pino, Mikel Sanz, David Pardo, and Jose I Aizpurua. Bayesian physics informed neural networks for reliable transformer prognostics. *arXiv*, 2025. 2

[47] Fuji Ren and Yanwei Bao. A review on human-computer interaction and intelligent robots. *International Journal of Information Technology & Decision Making*, 2020. 1

[48] Pengfei Ren, Chao Wen, Xiaozheng Zheng, Zhou Xue, Haifeng Sun, Qi Qi, Jingyu Wang, and Jianxin Liao. Decoupled iterative refinement framework for interacting hands reconstruction from a single rgb image. In *ICCV*, 2023. 1, 2

[49] Maximilian Rixner and Phaedon-Stelios Koutsourelakis. A probabilistic generative model for semi-supervised training of coarse-grained surrogates and enforcing physical constraints through virtual observables. *Journal of Computational Physics*, 2021. 2, 3, 5

[50] Javier Romero, Dimitris Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics*, 2017. 3, 4

[51] Soshi Shimada, Franziska Mueller, Jan Bednarik, Bardia Doosti, Bernd Bickel, Danhang Tang, Vladislav Golyanik, Jonathan Taylor, Christian Theobalt, and Thabo Beeler. Macs: Mass conditioned 3d hand and object motion synthesis. In *3DV*, 2024. 2

[52] Dule Shu, Zijie Li, and Amir Barati Farimani. A physics-informed diffusion model for high-fidelity flow field reconstruction. *Journal of Computational Physics*, 2023. 3

[53] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. *arXiv*, 2022. 6

[54] Tze Ho Elden Tse, Kwang In Kim, Ales Leonardis, and Hyung Jin Chang. Collaborative learning for hand and object reconstruction with attention-guided graph convolution. In *CVPR*, 2022. 1, 2

[55] Zhigang Tu, Zhisheng Huang, Yujin Chen, Di Kang, Linchao Bao, Bisheng Yang, and Junsong Yuan. Consistent 3d hand reconstruction in video via self-supervised learning. *IEEE TPAMI*, 2023. 6

[56] Congyi Wang, Feida Zhu, and Shilei Wen. Memahand: Exploiting mesh-mano interaction for single image two-hand reconstruction. In *CVPR*, 2023. 1, 2

[57] Kevin Xie, Tingwu Wang, Umar Iqbal, Yunrong Guo, Sanja Fidler, and Florian Shkurti. Physics-based human motion estimation and synthesis from videos. In *ICCV*, 2021. 4

[58] Pengfei Xie, Wenqiang Xu, Tutian Tang, Zhenjun Yu, and Cewu Lu. Ms-mano: Enabling hand pose tracking with biomechanical constraints. In *CVPR*, 2024. 2, 6

[59] Lixin Yang, Kailin Li, Xinyu Zhan, Jun Lv, Wenqiang Xu, Jiefeng Li, and Cewu Lu. Artiboost: Boosting articulated 3d hand-object pose estimation via online exploration and synthesis. In *CVPR*, 2022. 1, 2

[60] Zeshi Yang, Kangkang Yin, and Libin Liu. Learning to use chopsticks in diverse gripping styles. *ACM Transactions on Graphics (TOG)*, 2022. 2

[61] Ian T Young and Lucas J Van Vliet. Recursive implementation of the gaussian filter. *Signal processing*, 44(2):139–151, 1995. 3

[62] Zhengdi Yu, Shaoli Huang, Chen Fang, Toby P. Breckon, and Jue Wang. Acr: Attention collaboration-based regressor for arbitrary two-hand reconstruction. In *CVPR*, 2023. 1, 2

[63] Zongsheng Yue, Jianyi Wang, and Chen Change Loy. Resshift: Efficient diffusion model for image super-resolution by residual shifting. *NeurIPS*, 2023. 4

[64] Baowen Zhang, Yangang Wang, Xiaoming Deng, Yinda Zhang, Ping Tan, Cuixia Ma, and Hongan Wang. Interacting two-hand 3d pose and shape reconstruction from single color image. In *ICCV*, 2021. 1, 2

[65] Pengfei Zhang and Deying Kong. Handformer2t: A lightweight regression-based model for interacting hands pose estimation from a single rgb image. In *WACV*, 2024. 2

[66] Xiong Zhang, Qiang Li, Hong Mo, Wenbo Zhang, and Wen Zheng. End-to-end hand mesh recovery from a monocular rgb image. In *ICCV*, 2019. 2

[67] Yufei Zhang, Jeffrey O Kephart, Zijun Cui, and Qiang Ji. Physpt: Physics-aware pretrained transformer for estimating human dynamics from monocular videos. In *CVPR*, 2024. 2, 4, 5, 6

[68] Yufei Zhang, Jeffrey O Kephart, and Qiang Ji. Weakly-supervised 3d hand reconstruction with knowledge prior and uncertainty guidance. In *ECCV*, 2024. 1, 2

[69] Yufei Zhang, Zijun Cui, Jeffrey O Kephart, and Qiang Ji. Diffusion-based 3d hand motion recovery with intuitive physics. In *ICCV*, 2025. 2, 4, 6, 3

[70] Fuwei Zhao, Zhenyu Xie, Michael Kampffmeyer, Haoye Dong, Songfang Han, Tianxiang Zheng, Tao Zhang, and Xiaodan Liang. M3d-vton: A monocular-to-3d virtual try-on network. In *ICCV*, 2021. 1

[71] Keyang Zhou, Bharat Lal Bhatnagar, Jan Eric Lenssen, and Gerard Pons-Moll. Toch: Spatio-temporal object-to-hand correspondence for motion refinement. In *ECCV*, 2022. 1, 2

[72] Zhishan Zhou, Shihao Zhou, Zhi Lv, Minqiang Zou, Yao Tang, and Jiajun Liang. A simple baseline for efficient hand mesh reconstruction. In *CVPR*, 2024. 2, 6

[73] Andrea Ziani, Zicong Fan, Muhammed Kocabas, Sammy Christen, and Otmar Hilliges. Tempclr: Reconstructing hands via time-coherent contrastive learning. In *3DV*, 2022. 1, 2, 6

[74] Christian Zimmermann and Thomas Brox. Learning to estimate 3d hand pose from single rgb images. In *ICCV*, 2017. 2

[75] Binghui Zuo, Zimeng Zhao, Wenqian Sun, Wei Xie, Zhou Xue, and Yangang Wang. Reconstructing interacting hands with interaction prior from monocular images. In *ICCV*, 2023. 1, 2

PAD-Hand: Physics-Aware Diffusion for Hand Motion Recovery

Supplementary Material

The supplementary material is composed of an appendix and a video file.

Appendix

- Section A: Details of the Euler-Lagrange Terms
- Section B: Bayesian Inference: Derivations
- Section C: Additional Experiments
- Section D: Additional Visualizations
- Section E: Discussion: PAD-Hand vs. DIP

Video file

- **"Hand Motion Comparison With Variance.mp4"**: Hand motion comparison with image-based estimates from WiLoR [45] and includes our dynamic variance estimation visualizations at joint, mesh level.

A. Details of the Euler-Lagrange Terms

In this section, we further clarify the terms in Eq. 1. Generalized mass matrix $\mathtt{M}$ captures how the hand's mass and inertia are distributed across joints. It determines how joint accelerations are related to the required generalized forces. The term $\mathtt{C}$ represents Coriolis and centrifugal effects caused by motion, accounting for velocity-dependent interactions between different joints. The term $\mathtt{g}$ corresponds to gravitational forces, i.e., the torques induced by gravity under the current hand configuration. Finally, $\mathcal{F}$ denotes the net generalized forces that explain the observed hand motion, including the forces needed to produce the measured accelerations and to balance the dynamic effects captured by $\mathtt{M}$, $\mathtt{C}$, and $\mathtt{g}$.

B. Bayesian Inference: Derivations

In this section, we derive equations 14, 15 and 16. We start with our sampling equation

$$x_{1:T}^{n-1} = A_n x_{1:T}^n + B_n \hat{x}_{1:T} + \Sigma_n \epsilon \tag{18}$$

Taking variance on both sides leads to

$$\begin{aligned}\mathrm{Var}(x_{1:T}^{n-1}) &= A_n^2 \mathrm{Var}(x_{1:T}^n) + B_n^2 \mathrm{Var}(\hat{x}_{1:T}) \\ &+ \Sigma_n^2 + 2A_nB_n\mathrm{Cov}(x_{1:T}^n, \hat{x}_{1:T}) \\ &+ 2A_nC_n\mathrm{Cov}(x_{1:T}^n, \epsilon) + 2B_nC_n\mathrm{Cov}(\hat{x}_{1:T}, \epsilon)\end{aligned} \tag{19}$$

Since $\epsilon \sim \mathcal{N}(0, I)$ is independent of $x_{1:T}^n$ and $\hat{x}_{1:T}$, their covariances $\mathrm{Cov}(x_{1:T}^n, \epsilon)$, $\mathrm{Cov}(\hat{x}_{1:T}, \epsilon)$ are zero. Hence we have

$$\begin{aligned}\mathrm{Var}(x_{1:T}^{n-1}) &= A_n^2 \mathrm{Var}(x_{1:T}^n) + B_n^2 \mathrm{Var}(\hat{x}_{1:T}) \\ &+ \Sigma_n^2 + 2A_nB_n\mathrm{Cov}(x_{1:T}^n, \hat{x}_{1:T})\end{aligned} \tag{20}$$

Now we derive the covariance $\mathrm{Cov}(x_{1:T}^n, \hat{x}_{1:T})$ as follows:

$$\mathrm{Cov}(x_{1:T}^n, \hat{x}_{1:T}) = \mathbb{E}[x_{1:T}^n \cdot \hat{x}_{1:T}] - \mathbb{E}[x_{1:T}^n] \cdot \mathbb{E}[\hat{x}_{1:T}] \tag{21}$$

Using the law of total expectation $\mathbb{E}[\mathbb{E}[X|Y]] = \mathbb{E}[X]$, $\mathbb{E}[\hat{x}_{1:T}]$ can be written as

$$\mathbb{E}[\hat{x}_{1:T}] = \mathbb{E}_{x_{1:T}^n}[\mathbb{E}[\hat{x}_{1:T}|x_{1:T}^n]] = \mathbb{E}_{x_{1:T}^n}[\hat{X}_{1:T}] \tag{22}$$

where $\hat{X}_{1:T} = f_\phi(x_{1:T}^n, y_{1:T}, n)$ and for the first term

$$\begin{aligned}\mathbb{E}[x_{1:T}^n \cdot \hat{x}_{1:T}] &= \mathbb{E}_{x_{1:T}^n}[x_{1:T}^n \cdot \mathbb{E}[\hat{x}_{1:T}|x_{1:T}^n]] \\ &= \mathbb{E}_{x_{1:T}^n}[x_{1:T}^n \cdot \hat{X}_{1:T}]\end{aligned} \tag{23}$$

Integrating equations 22 and 23 to the equation 21 leads to

$$\begin{aligned}\mathrm{Cov}(x_{1:T}^n, \hat{x}_{1:T}) &= \mathbb{E}_{x_{1:T}^n}[x_{1:T}^n \cdot \hat{X}_{1:T}] \\ &- \mathbb{E}[x_{1:T}^n] \cdot \mathbb{E}_{x_{1:T}^n}[\hat{X}_{1:T}]\end{aligned} \tag{24}$$

which we approximate via Monte Carlo (MC) estimation:

$$\begin{aligned}\mathrm{Cov}(x_{1:T}^n, \hat{x}_{1:T}) &\approx \frac{1}{S}\sum_{i=1}^{S}(x_{1:T}^{n,i} \cdot \hat{X}_{1:T}^i) \\ &- \mathbb{E}[x_{1:T}^n] \cdot \frac{1}{S}\sum_{i=1}^{S}\hat{X}_{1:T}^i\end{aligned} \tag{25}$$

where S is the sample size.

For the equation 16, taking expectation on both sides of equation 18 leads to

$$\mathbb{E}[x_{1:T}^{n-1}] = A_n\mathbb{E}[x_{1:T}^n] + B_n\mathbb{E}[\hat{x}_{1:T}] + \Sigma_n\mathbb{E}[\epsilon] \tag{26}$$

Since $\epsilon \sim \mathcal{N}(0, I)$, $\mathbb{E}[\epsilon]$ is zero and by using law of total expectation on $\mathbb{E}[\hat{x}_{1:T}]$ we get

$$\mathbb{E}[x_{1:T}^{n-1}] = A_n\mathbb{E}[x_{1:T}^n] + B_n\mathbb{E}_{x_{1:T}^n}[\hat{X}_{1:T}] \tag{27}$$

C. Additional Experiments

C.1. Ablation Study on Data-driven Losses

In this section, we perform an ablation study on data-driven losses which are the backbone loss $\mathcal{L}_b$, geometric loss $\mathcal{L}_g$, and consistency regularization $\mathcal{L}_r$ and report the results in Table 5. $\mathcal{L}_g$ and $\mathcal{L}_r$ enable data-driven improvements, while further adding our physics loss yields the best performance.

Table 5. **Effectiveness of data-driven losses.**

Loss				Evaluation Metrics		
$\mathcal{L}_b$	$\mathcal{L}_g$	$\mathcal{L}_r$	$\mathcal{L}_{EL}$	PA-MPJPE↓	MPJPE↓	ACCEL↓
WiLoR				4.88	12.75	6.70
✓				5.01	12.57	3.83
✓	✓			4.83	10.89	3.57
✓	✓	✓		4.65	10.62	3.36
✓	✓	✓	✓	**4.63**	**10.56**	**3.34**

C.2. Refining Additional Baselines and Robustness to Initial Pose Estimates

Since PAD-Hand takes the predictions of a baseline model as input and refines them, it can be applied on top of different hand pose estimators. It is also robust to poor initial poses. To demonstrate this generality, we further evaluate PAD-Hand using the outputs of two recent baseline models, as shown in Tab. 6. In both cases, PAD-Hand consistently improves PA-MPJPE, MPJPE, and ACCEL, showing that our method serves as an effective refinement module across different baselines. We further test robustness by replacing 80% of a sequence with Gaussian noise, which increases the initial PA-MPJPE to 24.27. Under this severe corruption, our model reduces it to 6.53.

Table 6. **Comparison to more SOTAs on DexYCB.**

Method	PA-MPJPE↓	MPJPE↓	ACCEL↓
HaMeR [43]	4.70	16.71	7.14
HaMeR+**Ours**	**4.60**	**11.09**	**3.37**
HandOccNet [42]	5.80	14.0	9.03
HandOccNet+**Ours**	**5.31**	**10.95**	**3.38**

C.3. Statistical Significance Analysis

In this section, we verify that the performance gains obtained with $\mathcal{L}_{EL}$ are statistically significant. To this end, we train two model variants on DexYCB using 5 random seeds: one with $\mathcal{L}_{EL}$ and one without it. We then report the mean and standard deviation over the evaluation metrics. PAD-Hand achieves 4.61 ± 0.02, 10.57 ± 0.02, and 3.33 ± 0.01 for PA-MPJPE, MPJPE, and ACCEL, respectively, compared with 4.66 ± 0.01, 10.65 ± 0.02, and 3.36 ± 0.01 for the variant without the physics residual. These improvements are statistically significant under a paired t-test, with p-values of 0.001, 0.001, and 0.002, respectively.

C.4. Ablation Study on a Challenging Dataset

In this section, we evaluate the generalization ability of PAD-Hand and significance of $\mathcal{L}_{EL}$ by testing it on a dataset that was not used during training. For this, we employ a more challenging hands–object interaction dataset TACO [37], which contains diverse and complex manipulation sequences. The quantitative results in Table 7 show that PAD-Hand reduces PA-MPJPE from 8.37 mm to 8.02 mm and MPJPE from 25.13 mm to 24.38 mm. In addition, we observe a substantial improvement in acceleration error, which decreases from 5.47 mm/frame^2 to 1.84 mm/frame^2, indicating that our method produces smoother and more physically plausible motions in this challenging setting. Furthermore, using $\mathcal{L}_{EL}$ shows stronger generalization, improving PA-MPJPE by 0.35 compared to 0.13 with $\mathcal{L}_{data}$ alone (Tab. 7), ensuring physically grounded motion beyond the training distribution. In addition, when trained with only 10% of the DexYCB training data, adding this loss during WiLoR refinement effectively reduces performance degradation as shown in Table 8. The PA-MPJPE is 6.67 with $\mathcal{L}_{EL}$ and degrades to 7.90 without $\mathcal{L}_{EL}$, demonstrating $\mathcal{L}_{EL}$'s value as a physical regularizer under limited data.

Table 7. **Generalization to unseen TACO [37].** S1 testing split is used for evaluation.

Method	PA-MPJPE↓	MPJPE↓	ACCEL↓
WiLoR	8.37	25.13	5.47
$\mathcal{L}_{data}$	8.24	25.02	2.44
$\mathcal{L}_{data}$ + $\mathcal{L}_{EL}$ (**Ours**)	**8.02**	**24.38**	**1.87**

Table 8. **Data efficiency under $\mathcal{L}_{EL}$ on DexYCB.**

Method	PA-MPJPE↓	MPJPE↓
$\mathcal{L}_{data}$	7.90	27.52
$\mathcal{L}_{data}$ + $\mathcal{L}_{EL}$ (**Ours**)	**6.67**	**25.84**

C.5. Additional Results on Physical Plausibility

In this section, we further assess the effectiveness of our physics integration by computing the Euler-Lagrange residual as an additional metric of physical plausibility, defined as

$$\mathcal{R}(\mathtt{q}) = \frac{1}{T}\sum_{t=1}^{T} ||\mathtt{M}_t\ddot{\mathtt{q}} + \mathtt{C}_t + \mathtt{g}_t - \bar{\mathcal{F}}||_1 \tag{28}$$

where $\bar{\mathcal{F}}$ is the pseudoforce computed from ground-truth motion data. Unlike the acceleration error, which primarily captures temporal smoothness, $\mathcal{R}(\mathtt{q})$ directly measures consistency with our physics formulation and therefore offers a complementary perspective on motion quality. As reported in Table 9, our physics-aware integration achieves

Table 9. **Ablation on physics integration on DexYCB**. † denotes that values are scaled by 10^{-3}.

Loss	PA-MPJPE↓	MPJPE↓	ACCEL↓	†$\mathcal{R}(\mathtt{q})$ ↓
*WiLoR [45]	4.88	12.75	6.70	19.12
SmoothFilter [61]	4.77	12.82	**3.10**	14.33
$\mathcal{L}_{data}$	4.65	10.62	3.36	15.16
$\mathcal{L}_{data}$ + $\mathcal{L}_{EL}^{D}$	4.66	10.61	3.35	13.45
Ours	**4.63**	**10.56**	3.34	**9.04**

a lower residual than the deterministic counterpart, indicating that it produces motions that are more consistent with the underlying dynamics. Also, deterministic integration results in worse performance in PA-MPJPE (4.66 mm vs. 4.65 mm) in comparison to the data-driven approach (i.e., $\mathcal{L}_{data}$). For reference, we also include results for the SmoothFilter [61] which applies Gaussian smoothing to the WiLoR's predictions and can be viewed as a naive baseline that implicitly drives high-frequency temporal variations (e.g., acceleration) toward zero, without modeling uncertainty in the underlying image-based estimates. Notably, although SmoothFilter outperforms PAD-Hand in terms of acceleration error, reconstruction accuracy is far below than us indicating the need to have $\mathcal{R}(\mathtt{q})$ metric for a better comparison in terms of physical plausibility.

D. Additional Visualizations

In this section, we present additional visualizations on three representative HO3D sequences to further illustrate the effectiveness of PAD-Hand (Figure 6). Across all examples, our pipeline not only refines the jittery motions produced by the baseline but also reliably flags segments with rapid changes through elevated variance estimates. For instance, in sequence III (a), there is a sudden change in the thumb motion and the corresponding variance maps in sequence III (b) assign a higher variance to the thumb joint, indicating an increased dynamic uncertainty in this fast-motion regime. Similar patterns are observed in the other sequences, where motion discontinuities and high-frequency jitter are both smoothed in the refined trajectories and highlighted by increased variance, providing an interpretable indicator of challenging image-based estimates. We also provide video versions of these sequences in "Hand Motion Comparison With Variance.mp4" file.

E. Discussion: PAD-Hand vs. DIP

Closest to our work is the DIP model proposed by Zhang *et al.* [69], which also employs a conditional diffusion model for hand motion recovery. However, our approach differs from DIP in several important aspects:

- DIP is trained with synthetic noise by adding Gaussian noise to the ground-truth motion, whereas PAD-Hand takes the predictions of an image-based estimator as input during training.
- DIP additionally outputs hand shape parameters and is supervised with a shape loss, while in our pipeline the shape parameters are kept fixed and we focus on refining the pose parameters.

Because of these design differences, a strictly fair comparison is not possible. Nevertheless, we report DIP results and use their officially released models for evaluation. Note that evaluating $\mathcal{R}(\mathtt{q})$ on HO3D is not feasible, since pose parameters for the test set are not publicly available. As shown in Tables 10 and 11, PAD-Hand consistently outperforms DIP in terms of reconstruction accuracy. Although DIP achieves a lower acceleration error on DexYCB (Table 10), its Euler-Lagrange residual $\mathcal{R}(\mathtt{q})$ remains higher than ours, indicating that our trajectories are better aligned with the underlying dynamics.

Table 10. **Comparison to DIP [69] on DexYCB**. † denotes that values are scaled by 10^{-3}.

Loss	PA-MPJPE↓	MPJPE↓	ACCEL↓	†$\mathcal{R}(\mathtt{q})$ ↓
*WiLoR [45]	4.88	12.75	6.70	19.12
WiLoR + DIP [69]	4.81	12.88	**3.23**	13.22
WiLoR + **Ours**	**4.63**	**10.56**	3.34	**9.04**

Table 11. **Comparison to DIP [69] on HO3D**.

Loss	PA-MPJPE↓	ACCEL↓
*WiLoR [45]	7.50	4.98
WiLoR + DIP [69]	7.55	3.03
WiLoR + **Ours**	**7.43**	**2.71**

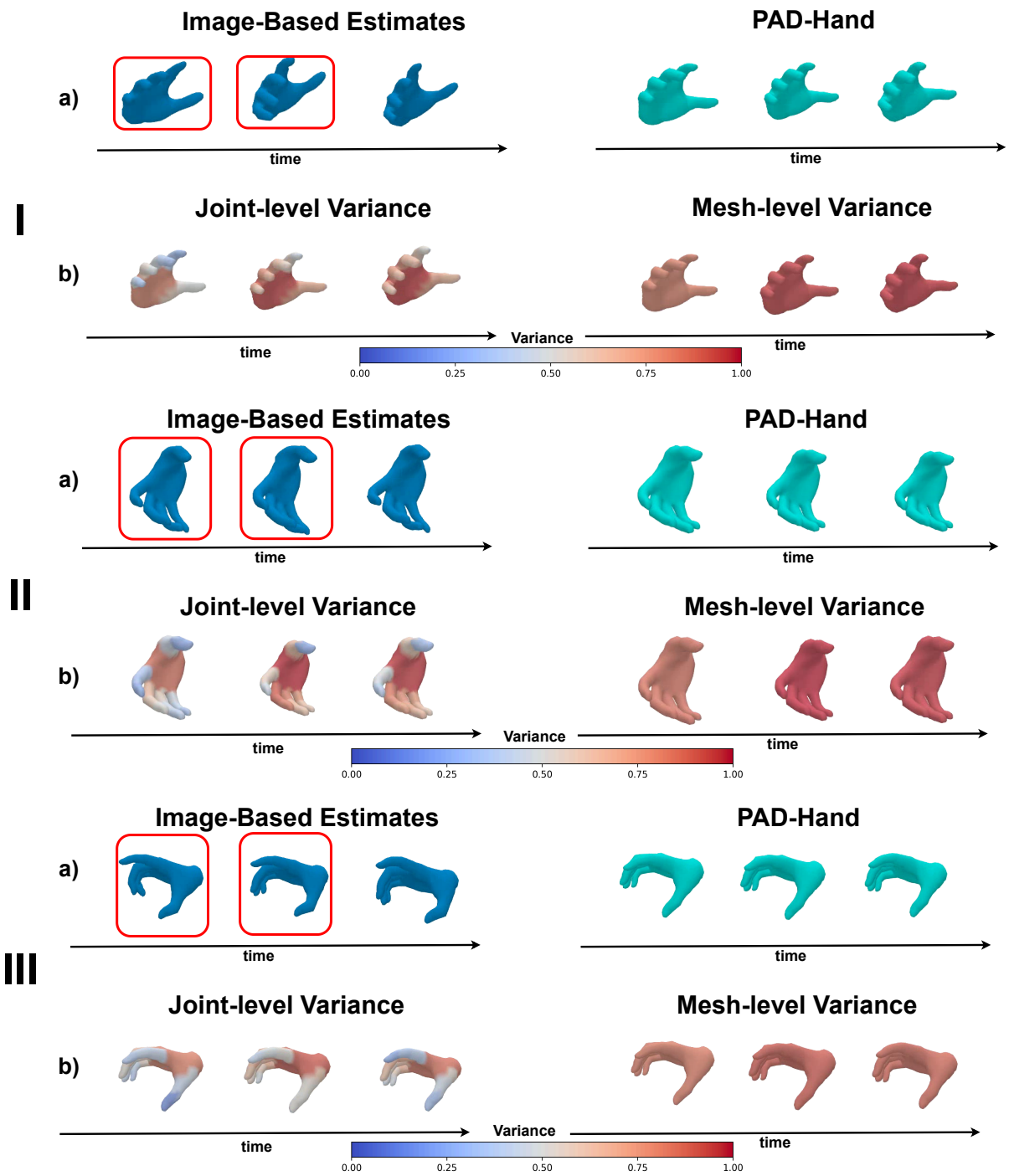

Figure 6. **Refined motion estimates by PAD-Hand with dynamic variance on HO3D.** We visualize three representative sequences (I–III). In each block, row (a) compares the original image-based motion estimates to the trajectories refined by PAD-Hand, while row (b) shows the corresponding variance estimations in terms of joint-level and mesh-level dynamic variance. The red boxes highlight frames where the image-based estimates exhibit strong jitter.