UniTac2Pose: A Unified Approach Learned in Simulation for Category-level Visuotactile In-hand Pose Estimation

Mingdong Wu [1,2*], **Long Yang** [1,2*], **Jin Liu** [3,4], **Weiyao Huang**[1,2], **Lehong Wu**[1,2], **Zelin Chen**[3,4], **Daolin Ma**[3,4], **Hao Dong**[1,2 †]
[1] Center on Frontiers of Computing Studies, School of Computer Science, Peking University
[2] PKU-AgiBot Lab, [3] School of Ocean and Civil Engineering, Shanghai Jiao Tong University,
[4] Xense Robotics

Abstract: Accurate estimation of the in-hand pose of an object based on its CAD model is crucial in both industrial applications and everyday tasks, ranging from positioning workpieces and assembling components to seamlessly inserting devices like USB connectors. While existing methods often rely on regression, feature matching, or registration techniques, achieving high precision and generalizability to unseen CAD models remains a significant challenge. In this paper, we propose a novel three-stage framework for in-hand pose estimation. The first stage involves sampling and pre-ranking pose candidates, followed by iterative refinement of these candidates in the second stage. In the final stage, post-ranking is applied to identify the most likely pose candidates. These stages are governed by a unified energy-based diffusion model, which is trained solely on simulated data. This energy model simultaneously generates gradients to refine pose estimates and produces an energy scalar that quantifies the quality of the pose estimates. Additionally, borrowing the idea from the computer vision domain, we incorporate a render-compare architecture within the energy-based score network to significantly enhance sim-to-real performance, as demonstrated by our ablation studies. We conduct comprehensive experiments to show that our method outperforms conventional baselines based on regression, matching, and registration techniques, while also exhibiting strong intra-category generalization to previously unseen CAD models. Moreover, our approach integrates tactile object pose estimation, pose tracking, and uncertainty estimation into a unified framework, enabling robust performance across a variety of real-world conditions. The polished video demonstrations and detailed appendix can be found at the anonymous website https://unitac2pose-web.vercel.app/.

Keywords: Tactile Pose Estimation, Diffusion Model, Precise Manipulation

1 Introduction

Precise pick-and-place operations are essential in industrial applications and daily tasks, from positioning workpieces and assembling components to inserting devices like USB connectors. This requires the robot to localize the objects' in-hand pose based on its CAD model with high precision [1]. A promising approach estimates the in-hand pose using high-resolution tactile sensors, which provide rich geometrical contact information and are highly discriminative [1]. Unlike vision-based methods [2], tactile-based approaches are immune to extrinsic calibration errors or occlusions.

Robotic in-hand object pose estimation is a challenging task due to several factors: the need to handle previously unseen objects, the requirement to regrasp when the estimated pose has high uncertainty [3], and the necessity to track pose changes caused by extrinsic collisions with the environment. A common approach is to train a pose regressor using tactile images and the object's CAD

*: equal contribution, †: corresponding author

9th Conference on Robot Learning (CoRL 2025), Seoul, Korea.

model on a simulated dataset. However, due to the localized nature of tactile sensing, this often leads to significant ambiguity—multiple poses can result in similar tactile imprints. Classical registration methods such as ICP and FilterReg [4, 5, 6] attempt to align tactile point clouds with the global object model. Yet, these methods are highly sensitive to initialization and prone to local minima. Moreover, tactile-based depth estimates are inherently noisy due to the monocular nature of most sensors, further degrading performance. Another line of work [1, 5, 3] explores feature-matching strategies. For example, Tac2Pose [4] converts tactile RGB images into binary contact maps and matches them to rendered maps from pose grids using features learned via MoCo [7]. However, these maps are noisy, often object-specific, and fail to capture in-plane surface features—such as a triangular ridge on a flat surface, especially when the object is under significant deformation.

We propose UniTac2Pose, a unified framework for in-hand object pose estimation, tracking, and uncertainty estimation, as illustrated in Figure 1. At its core is an energy-based diffusion model that estimates the (unnormalized) log-likelihood of an object pose, conditioned on observed tactile imprints and the object's CAD model. This scalar energy score enables candidate pose ranking, while its gradient indicates the optimal direction for refinement. The framework consists of three stages: pre-ranking, refinement, and post-ranking. During inference, we first sample initial pose candidates from a prior distribution, such as the one in [2], and pre-rank them using the energy network. Next, the top candidates are refined through gradient-based optimization guided by the energy network. Finally, a post-ranking stage selects the most likely pose. The energy network is trained end-to-end using a score-matching objective on purely simulated data. To bridge the sim-to-real gap, we adopt a render-and-compare architecture that significantly improves performance, as validated by our ablation studies.

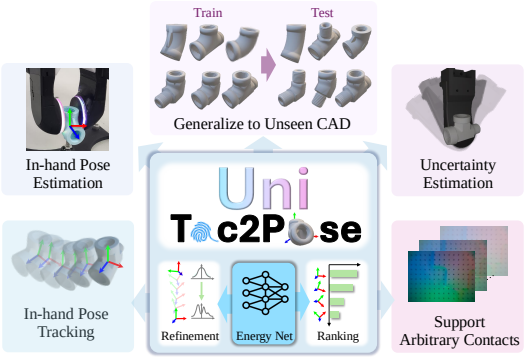


Figure 1: The core of UniTac2Pose is an energy-based diffusion model that unifies tactile pose estimation, tracking, and uncertainty, conditioned on multi-contact and generalizable to unseen CADs.

Our approach offers several key advantages: **Unified functionality.** UniTac2Pose integrates pose estimation, tracking, and uncertainty quantification. For example, pose tracking is achieved by centering the initial pose prior on the previous prediction, while pose uncertainty is quantified via the variance of refined pose candidates. **End-to-end training.** Unlike prior methods that rely on feature matching or intermediate regressors, our method directly optimizes for pose likelihood using end-to-end training, avoiding compounding errors. **Generalization to unseen intra-category objects.** By conditioning on multiple tactile imprints and leveraging simulated training, our framework generalizes effectively to unseen CAD models without requiring real-world data.

We validate our method through extensive real-world experiments. UniTac2Pose consistently outperforms regression-, matching-, and registration-based baselines in both pose estimation and tracking tasks, and demonstrates strong generalization across object categories.

In summary, our contributions are as follows:

- We introduce UniTac2Pose, a unified framework for visuotactile in-hand object pose estimation, tracking, and uncertainty quantification, capable of handling diverse contact scenarios and generalizing to unseen objects.
- We propose a novel three-stage framework based on an energy-based diffusion model trained solely on simulated data, requiring no real-world supervision.
- Our method achieves state-of-the-art performance in real-world pose estimation and tracking, and to the best of our knowledge, is the first to demonstrate intra-category generalization in visuotactile pose estimation.

2 Related Works

2.1 Tactile Object Pose Estimation

Tactile perception has evolved significantly, from early work using low-resolution, often planar or bulky sensors [8, 9, 10, 11, 12, 13, 14, 15], to modern high-resolution tactile sensing for more refined pose estimation. Early approaches often relied on binary contact signals and required repeated readings [16, 17], limiting robustness and efficiency. Subsequent methods combined vision and tactile sensing to improve global pose estimation [18, 19, 20, 21, 22]. However, in many of these, tactile input served merely as a binary signal to refine visual estimates, inherently bounding accuracy to the visual modality. In contrast, recent efforts focus on using high-resolution tactile data as a primary modality, enabling contact shape recovery [23, 24, 25], or leveraging deformable tactile surfaces for improved localization [26]. Distributed tactile arrays have also been explored to enhance in-hand object tracking [27, 28].

Our work builds on GelSlim [29], a high-resolution visual-tactile sensor previously applied in grasp assessment [30], 3D shape reconstruction [31], and tactile-based manipulation [32]. Registration-based pose estimation methods using tactile point clouds [4, 33] have shown promise, but remain sensitive to initialization and are affected by depth noise inherent in tactile data. The most relevant prior line of work is Tac2Pose and its variants [34, 1, 3], which formulate pose estimation as a probabilistic inference problem. Our method differs fundamentally by learning a unified energy-based model in an end-to-end fashion, avoiding the need to chain pose refinement steps and enabling category-level generalization to unseen object instances via CAD model conditioning.

2.2 Applications of Score-based Generative Models

Score-based generative models have emerged as powerful tools for modeling complex data distributions via gradient estimation of log-likelihoods [35, 36]. Denoising score matching (DSM)[36] was introduced to improve training stability, followed by sliced score matching[37], annealed training [38], and other training refinements [39]. These advances culminated in diffusion-based models that perform particularly well in high-dimensional domains such as image synthesis [40], leveraging continuous-time stochastic processes for generation.

More recent efforts have broadened the applicability of these models to 3D and robotics tasks, including point cloud generation [41], denoising [42], depth completion [43], and human pose estimation [44]. Notably, works such as GenPose [45] have demonstrated how score-based and energy-based models can be integrated for robust 6D pose estimation under uncertainty, using diffusion sampling guided by learned energy fields. In contrast, our approach trains a single, unified energy network that simultaneously computes gradients for iterative pose refinement and outputs scalar energy values to assess pose quality. Additionally, we condition this network on object CAD models, enabling render-and-compare strategies for instance-specific generalization—a key difference from category-level-only approaches such as GenPose.

3 Method

Problem Statement: This work addresses the task of estimating the 6D pose of in-hand objects within the Tool Center Point (TCP) frame, crucial for robotic pick-and-place operations. Given a CAD model of an object O, we aim to estimate the 6D in-hand pose $\boldsymbol{p} \in \mathbb{R}^{3+6}$ from tactile imprints $T = (T_1, T_2, \ldots, T_k)$, where k is the number of tactile contacts. We represent the pose as a 9-D variable $[R|T]$ following [45], with $R \in \mathbb{R}^6$ for rotation and $T \in \mathbb{R}^3$ for translation. To handle discontinuities in quaternion and Euler angle representations, we use a continuous 6-D rotation representation $[R_x|R_y]$ [46, 47].

Overview: We first generate a synthetic dataset using the FEM-based tactile simulator XENSIM, represented as $\mathcal{D} = \{(\boldsymbol{p}_i, O_i, T_i)\}_{i=1}^n$, where $\boldsymbol{p}_i \in \mathrm{SE}(3)$ denotes the 6D pose, and $O_i \in \mathbb{R}^{3\times N}$ is the canonical object point cloud. The tactile observations $T \in \mathbb{R}^{3\times H\times k\times W}$ are concatenated tactile images from GelSlim 3.0 [29], with H and W representing the image height and width.

In Sec. 3.2, we describe the training of an energy-based diffusion model E_θ on the synthetic dataset. During inference (Sec. 3.3), given a CAD model O^* and tactile imprints T^*, we estimate the 6D pose in three stages. First, initial pose candidates are sampled from a prior distribution [2], and

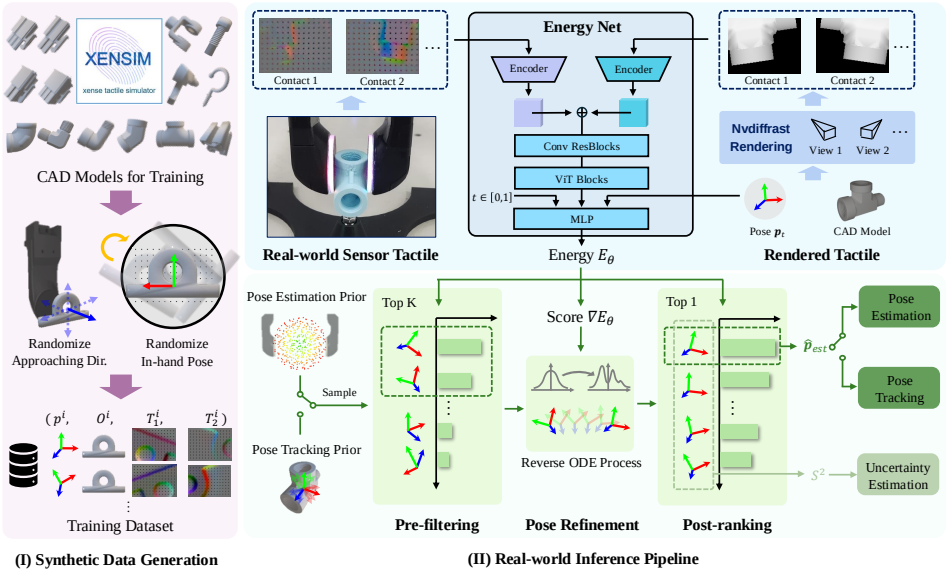


Figure 2: **Method Overview.** (I): We first generate a synthetic dataset using the FEM-based tactile simulator XENSIM. We randomly sample in-hand poses to generate a diverse training dataset with pure simulation. (II): During inference, the Energy Net takes real-world tactile, rendered tactile, object pose and diffusion timestep as inputs, and outputs the energy and score of the pose. For pose estimation and tracking, we sample N pose candidates from a prior distribution, and get the final pose by pre-filtering, refinement and post-ranking. For uncertainty estimation, we calculate the variance of refined poses to represent the uncertainty of the grasp.

low-likelihood candidates are discarded, leaving $\hat{\boldsymbol{p}}_1, \hat{\boldsymbol{p}}_2, ..., \hat{\boldsymbol{p}}_K$, with $K = 16$. These candidates are then refined using gradients from the energy network E_θ. Finally, the candidates are ranked by energy, and the one with the lowest energy is selected as the final pose. Additionally, our framework also supports pose tracking and uncertainty estimation (Appendix D).

3.1 Synthesizing Tactile Pose Estimation Dataset

We synthesize the tactile pose estimation dataset $\mathcal{D} = \{(\boldsymbol{p}_i, O_i, T_i)\}_{i=1}^n$ for each training object via simulating the RGB tactile image under different object-to-hand poses (*i.e.*, grasp poses) in a FEM-based tactile simulation. Due to the page limit, we defer data generation details to Appendix A.

3.2 Training the Energy-based Diffusion Model

UniTac2Pose is a three-stage framework unified by an energy-based diffusion model E_θ, trained on the synthesized dataset $\mathcal{D} = \{(\boldsymbol{p}_i, O_i, T_i)\}_{i=1}^n$ as described above. The energy network $E_\theta : \mathbb{R}^{3+6} \times \mathbb{R}^{3\times N} \times \mathbb{R}^{3\times H\times k\times W} \times [0,1] \to \mathbb{R}^1$ takes as input an intermediate pose variable $\boldsymbol{p}$, the canonical object point cloud O, the concatenated tactile images T and a continuous time variable $t \in [0,1]$, and outputs a scalar energy value $E_\theta(\boldsymbol{p}, O, T, t) \in \mathbb{R}^1$.

Training Objective: We assume the synthetic dataset $\mathcal{D}$ is sampled from an implicit data distribution $\mathcal{D} = \{(\boldsymbol{p}_i, O_i, T_i) \sim p_{\text{data}}(\boldsymbol{p}, O, T)\}$. The energy network is trained to match between the derivative of its output $\nabla_{\boldsymbol{p}} \log E_\theta(\boldsymbol{p}, O, T, t)$ regarding the input pose variable and the *score function* of the perturbed conditional distribution $\nabla_{\boldsymbol{p}} \log p_t(\boldsymbol{p}|O, T)$, for all $t \in [0,1]$, using the score-matching training objective [38, 37, 40].

The energy network E_θ is then trained via the following Denoising Score Matching (DSM) [36] objective:

$$\mathcal{L}(\theta) = \mathbb{E}_t \left\{ \lambda(t) \mathbb{E} \left[\left\| \nabla_{\boldsymbol{p}} E_\theta(\boldsymbol{p}(t), t, O, T) - \frac{\boldsymbol{p}(0) - \boldsymbol{p}(t)}{\sigma(t)^2} \right\|_2^2 \right] \right\} \tag{1}$$

where $\boldsymbol{p}(0) \sim p_{\text{data}}(\boldsymbol{p}(0)|O, T)$, $\boldsymbol{p}(t) \sim \mathcal{N}(\boldsymbol{p}(t); \boldsymbol{p}(0), \sigma^2(t)\mathbf{I})$, and $\epsilon = 10^{-5}$ is a hyper-parameter that denotes the minimal noise level.

Architectural Design: Instead of directly encoding all inputs using MLPs, CNNs, or PointNet, we introduce a key design, *i.e.*, render-compare, into the energy network E_θ, inspired by the impressive

results of FoundationPose [2]. During inference, the pose variable $\boldsymbol{p}(t)$ is first normalized to an SE(3) pose $\overline{\boldsymbol{p}}(t)$. Next, in the TCP frame, the canonical CAD model of the object is initialized at $\overline{\boldsymbol{p}}(t)$, and RGB images $I = (I_1, I_2, \ldots, I_K)$ are rendered from multiple pre-calibrated cameras using Nvdiffrast [48]. Note that the camera poses are based on pre-calibrated real sensor cameras. Unlike FoundationPose, we encode the horizontally concatenated tactile images and rendered images using two separate encoders, as they come from different modalities. Similar to FoundationPose, the features extracted by these CNNs are concatenated and subsequently fed into Convolutional Residual Blocks and ViT [49] Blocks. We adopt a parameterization trick, proven effective for training energy-based diffusion models in [45, 50], as follows:

$$E_\theta(\boldsymbol{p}, O, T, t) = \langle \boldsymbol{\Phi}_\theta(\boldsymbol{p}, O, T, t), \boldsymbol{p} \rangle \tag{2}$$

where $\boldsymbol{\Phi}_\theta : \mathbb{R}^{3+6} \times \mathbb{R}^{3\times N} \times \mathbb{R}^{3\times H\times k\times W} \times [0,1] \to \mathbb{R}^{3+6}$ is the main backbone of the energy network, as described above, and outputs a 9-dimensional vector, which matches the dimension of the pose variable.

3.3 Tactile In-hand Pose Estimation Using the Energy Network

According to [36], minimizing the objective in Eq. 1 leads to an optimal energy network E^*_θ that satisfies the following equation under some mild assumptions:

$$\nabla_{\boldsymbol{p}} E^*_\theta(\boldsymbol{p}, O, T, t) = \nabla_{\boldsymbol{p}} \log p_t(\boldsymbol{p}|O, T) \implies E^*_\theta(\boldsymbol{p}, O, T, t) = \log p_t(\boldsymbol{p}|O, T) + C(O, T) \tag{3}$$

where $C(O, T)$ is a constant independent of $\boldsymbol{p}$. Although the optimal energy model differs from the ground truth likelihood by the constant $C(O, T)$, it still serves as a reliable surrogate likelihood estimator for ranking candidates, given fixed tactile observations T and the object CAD model O:

$$E^*_\theta(\boldsymbol{p}_i, O, T, \epsilon) > E^*_\theta(\boldsymbol{p}_j, O, T, \epsilon) \iff \log p_\epsilon(\boldsymbol{p}_i|O, T) \quad > \log p_\epsilon(\boldsymbol{p}_j|O, T) \tag{4}$$

Motivated by these results, we propose a three-stage framework that utilizes the energy network's output for pose ranking in both pre-filtering and post-ranking and the gradient from the energy network for the pose refinement stage.

Pre-filtering Our framework begins by sampling a large set of initial pose candidates from a prior distribution π_{est}. For pose estimation, we use the prior distribution from [2] for global pose sampling. Specifically, we first uniformly sample N_s viewpoints from an icosphere centered on the object, with the camera facing the center, and then convert these camera-to-object poses into object-to-TCP poses. Next, using the trained energy model, we rank the candidates $\{\tilde{\boldsymbol{p}}_i\}_{i=1}^M$ into a sequence $\tilde{\boldsymbol{p}}_1 \succ \tilde{\boldsymbol{p}}_2 \cdots \succ \tilde{\boldsymbol{p}}_M$, where:

$$\tilde{\boldsymbol{p}}_i \succ \tilde{\boldsymbol{p}}_j \iff E_\theta(\tilde{\boldsymbol{p}}_i, O, T, \epsilon) > E_\theta(\tilde{\boldsymbol{p}}_j, O, T, \epsilon) \tag{5}$$

and M is a hyperparameter. We then filter out the bottom $M - K$ of candidates, leaving $\{\tilde{\boldsymbol{p}}_i\}_{i=1}^K$ as the output of the pre-filtering stage, where K is another hyperparameter.

Pose Refinement Further, we iteratively refine the candidates using gradients derived from the trained energy network. Specifically, we refine $\{\hat{\boldsymbol{p}}_i\}_{i=1}^K$ via a modified version of the *Probability Flow* (PF) ODE [40], integrating from $t = t_0$ to ϵ, to obtain the refined candidates $\{\hat{\boldsymbol{p}}_i\}_{i=1}^K$:

$$\frac{d\boldsymbol{p}}{dt} = -\sigma(t)\dot{\sigma}(t)\nabla_{\boldsymbol{p}} \log p_t(\boldsymbol{p}|O) \tag{6}$$

Here, the ODE starts from the initial pose candidates $\{\tilde{\boldsymbol{p}}_i\}_{i=1}^K$, where t_0 is a hyperparameter. The score function $\nabla_{\boldsymbol{p}} \log p_t(\boldsymbol{p}|O)$ is approximated by the gradient of the energy network $E_\theta(\boldsymbol{p}, O, T, t)$, and the ODE is solved using the RK45 solver [51]. Since the initial candidates are already close to the high-density regions of $p_{\text{data}}(\boldsymbol{p}|O, T)$, we start the refinement from a smaller t, where the gradient is more informative.

Post-Ranking Finally, we rank the refined pose candidates $\{\hat{\boldsymbol{p}}_i\}_{i=1}^K$ using the energy network, similar to the pre-filtering stage. Unlike GenPose [45], which aggregates predictions from multiple candidates, we select the single candidate with the highest energy as the final pose estimate:

$$\hat{\boldsymbol{p}}_{est} = \arg\max_i E_\theta(\hat{\boldsymbol{p}}_i, O, T, \epsilon) \tag{7}$$

3.4 Extending the Framework for Pose Tracking and Uncertainty Estimation

Our framework can be extended to support pose tracking and uncertainty estimation, enhancing object manipulation robustness. Due to the page limit, we defer the complete derivations and discussions to Appendix D. The key ideas are as follows:

Pose Tracking. We track the in-hand object pose by modifying the prior distribution $\pi_{\text{track}}(\tilde{\boldsymbol{p}}_{t+1})$ in the pre-filtering stage, incorporating temporal continuity from previous poses. This method enables real-time tracking at 10 Hz with fewer refinement steps.

Uncertainty Estimation. Uncertainty is estimated by computing the variance S^2 of the pose candidates $\{\hat{\boldsymbol{p}}_i\}_{i=1}^{K}$. The grasp with the lowest uncertainty is selected for re-grasping. Relative uncertainty between grasps is determined by comparing the S_i^2 values, as shown in Eq. 5.

4 Results

4.1 Experiment Setup

Objects and Synthetic Dataset. We generate synthetic and real-world evaluation datasets using 30 objects from the McMaster dataset. Following [34, 52], all objects are sourced from McMaster. We evaluate our method against baselines on 10 distinct objects, including symmetric shapes and those with localized features prone to inducing singularities. For each object, we generate 20,000 data points. To assess category-level generalization, we include two common industrial assembly categories: *pipe* and *connector*. The *pipe* category consists of 13 objects, with 8 for training and 5 for testing. The *connector* category consists of 7 objects, with 5 for training and 2 for testing. For each training object in *pipe*, we generate 10,000 data points, while for each training object in *connector*, we generate 5,000 data points. Each category contains structurally similar objects, allowing the network to learn shared features from the training set and generalize to unseen objects.

Real World Evaluation. Following [1], we use the GelSlim 3.0 for both simulation and real-world data collection. The hardware setup is detailed in Appendix B. For each object, we first collect a fixed number of grasps, recording both the tactile images and the robot's TCP pose. To obtain the ground truth object pose in the TCP frame, we replicate the real-world scene in simulation, manually annotate the first few ground-truth poses, and automatically generate the rest. We defer the detailed procedure to Appendix B. In total, we collect over 3800 grasps across all 30 objects.

Since the objects in our study include symmetric shapes, which introduce ambiguity in pose estimation (as multiple configurations of symmetric objects can appear identical), we use ADD-S as the evaluation metric. ADD-S is an adaptation of the ADD (Average Distance of Model Points) metric. Specifically, we apply ADD-S for objects with symmetry and ADD for objects without symmetry. For simplicity, we refer to this unified metric as ADD-S in our experiments.

$$\text{ADD-S} = \frac{1}{|\mathcal{M}|} \sum_{\mathbf{p} \in \mathcal{M}} \min_{\mathbf{q} \in \mathcal{M}} \|(\mathbf{R}_{\text{est}}\mathbf{p} + \mathbf{t}_{\text{est}}) - (\mathbf{R}_{\text{gt}}\mathbf{q} + \mathbf{t}_{\text{gt}})\| \tag{8}$$

As shown in Eq. 8, ADD-S addresses this challenge by considering the closest matching point on the ground-truth object when calculating distances, rather than relying on a one-to-one correspondence.

Baselines. Nevertheless, we implement 3 typical approaches from previous studies, *i.e.*, *FilterReg*, *Vanilla Regression*, and *Matching (Tac2Pose [1])* with same network architecture and capabilities to ensure a fair comparison. Implementation details are deferred to the Appendix. C.

4.2 Generalizable Tactile Pose Estimation within a Single Instance

We evaluate UniTac2Pose for tactile pose estimation at the instance level, training on synthesized datasets of a single object and testing on real-world tactile data from the same object, with potentially different grasps. Objects from McMaster, such as the *Cable Clip* and *Cotter*, as well as symmetrical objects like *Bear Housing* and *Round Nut*, are used.

Table 1: **Instance-level sim-to-real evaluation results evaluation results.** We report ADD-S (mm) and ADD (mm) errors for symmetric and non-symmetric objects respectively. Lower ADD/ADD-S error implies better performance. We compare our method with FilterReg, regression, and matching as baselines. We also conduct ablation studies on our key designs, i.e., using depth as rendered input, unifying score and energy network, render-compare, and RGB augmentation.

	Ours		Ablations			Baselines			Oracle
	RGB	Depth	w/o Unify	w/o R-C	w/o Aug	FilterReg(Global)	Regression	Matching	FilterReg(Partial)
Bear Housing	**2.5**	2.7	8.7	5.3	8.7	6.9	10.8	13.9	2.2
Cable Holder	2.7	**2.0**	7.8	9.9	3.5	7.8	3.1	12.9	2.8
Cable Clip	**1.6**	1.9	2.2	1.9	**1.6**	8.2	10.4	12.7	2.2
Round Nut	**2.4**	3.6	3.3	43.2	2.5	5.4	45.4	8.4	2.2
Cotter	**1.5**	1.8	4.1	7.5	18.1	21.0	3.9	30.2	3.5
Hook	**3.0**	9.7	3.4	29.9	32.7	20.7	35.2	36.9	2.7
Hose	1.5	1.2	**1.0**	3.2	1.4	20.7	1.8	13.9	1.6
Hydraulic	2.4	**2.3**	2.8	3.5	2.7	10.0	5.3	13.3	3.2
Stud	**2.2**	9.7	3.6	18.3	9.9	26.1	8.0	12.4	6.9
Rail	1.5	1.8	**1.1**	2.5	1.8	5.9	6.1	8.0	4.4
Average	**2.1**	3.7	3.8	12.5	8.3	13.3	13.0	16.3	3.2

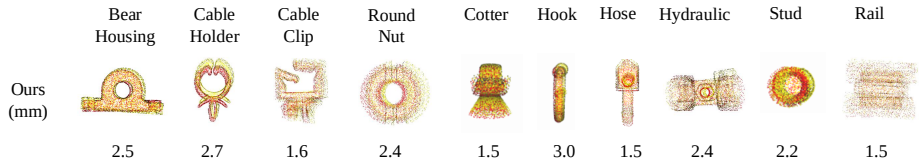
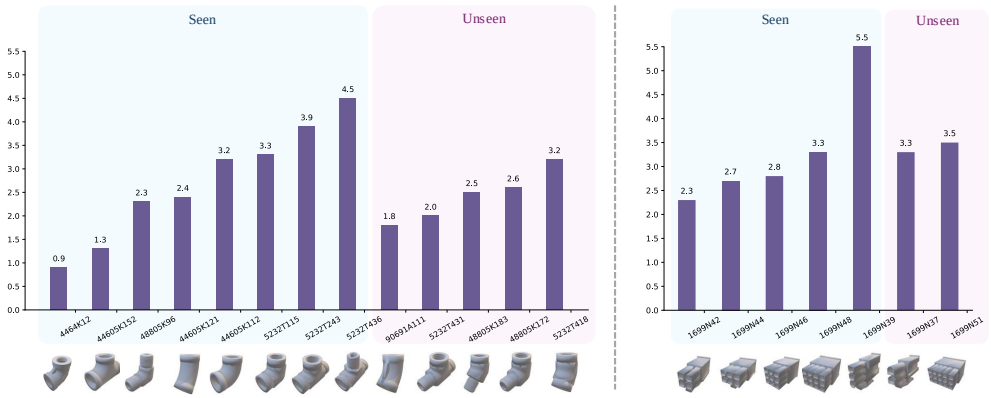


Figure 3: **Visualization of ADD errors.** We visualize point clouds of objects with ground truth poses (red) and estimated poses (yellow). The ADD errors are the same as reported in Tab. 1.

Comparison with Baselines. As shown in Tab.1, our method outperforms all baselines across all objects, demonstrating superior performance in real-world tactile pose estimation. While the *Regression* baseline works well for small objects with distinct features, it struggles with symmetric or larger ones. *FilterReg (Global)* fails on nearly all objects due to global point cloud registration issues, whereas *FilterReg (Partial)* performs comparably to our method. The *Matching* baseline underperforms due to discrepancies between simulation and real-world contact shapes, similar to issues in Tac2Pose[1]. Our approach, powered by score-matching training [45], handles these challenges effectively, providing robust predictions even with ambiguous or symmetric tactile data.

Ablation Studies. We compare our method against several ablations: removing the render-compare mechanism (*Ours w/o R-C*), training a separate score-based diffusion network for pose refinement instead of using the unified approach (*Ours w/o Unify*), and disabling data augmentation (*Ours w/o Aug(mentation)*). As shown in Tab. 1, our method consistently outperforms all of these variants. The most significant performance drop occurs when the render-compare mechanism is removed, highlighting its crucial role. Disabling data augmentation leads to a performance decline, especially for objects like *Cotter* and *Hook*, while other objects are less impacted. The variant without the unified approach (*Ours w/o Unify*) performs similarly to the full model but shows slightly higher ADD-S errors, demonstrating the importance of our unified energy network in effectively aligning the pose refinement, ranking, and evaluation stages. We also conduct further ablations on the three stages (i.e., pre-filtering, refinement, and post-ranking) and the choice of hyperparameters in Appendix F and Appendix G, respectively.

Robustness of the Render-Compare Modality. We evaluate the robustness of two alternative energy network versions that render RGB images and depth maps. Both yield comparable performance, but RGB-based rendering outperforms depth-based rendering for objects like *Stud* and *Hook*, suggesting it is more stable in tactile pose estimation.



(a) The Pipe Category

(b) The Connector Category

Figure 4: **Category-level sim-to-real evaluation accuracy.** For the pipe class, we train on the first 8 objects and evaluate all 13 objects. For the connector class, we train on the first 5 objects and evaluate all 7 objects. We report ADD-S (mm) and ADD (mm) errors for symmetric and non-symmetric objects, respectively. Lower ADD/ADD-S error implies better performance.

4.3 Generalizable Tactile Pose Estimation to Unseen Intra-category Objects

Having validated our method for instance-level pose estimation, we next assess its ability to generalize to unseen objects after training on multiple objects from a single category. To this end, we conduct a proof-of-concept experiment on two categories: *Pipe* and *Connector*, training on 8 and 5 objects, respectively, and testing on 5 and 2 objects. Despite training on a small dataset, our energy network learns generalizable representations due to shared local features within each category. As shown in Fig. 4, performance on unseen instances does not degrade significantly, suggesting that our method can generalize to new instances within a category and has the potential for broader generalization with large-scale training.

4.4 Validation of Uncertainty Estimation, Pose Tracking, and Arbitrary Contacts

We also validate the capabilities of our framework in uncertainty estimation, pose tracking, and handling arbitrary contact scenarios. Our experiments demonstrate that the framework effectively supports in-hand pose tracking with minimal error and stable frame rates. Additionally, our uncertainty estimation method consistently outperforms baseline approaches, and the system can generalize well to situations involving arbitrary subsets of tactile observations. Due to the page limit, we defer the detailed experimental results and additional discussions to the Appendix E.

5 Conclusion

In this work, we investigate tactile in-hand object pose estimation, a crucial task for high-precision pick-and-place manipulation. While existing approaches predominantly rely on regression, feature matching, or registration techniques, achieving high accuracy while maintaining adaptability to unseen CAD models remains a significant challenge. We propose a novel three-stage framework: the first stage involves sampling and pre-ranking pose candidates, followed by iterative refinement of these candidates in the second stage. In the final stage, post-ranking is applied to determine the most probable pose. All stages are governed by a unified energy-based diffusion model integrated with a render-compare architectural design, trained solely on simulated data. This approach alleviates the laborious and expensive process of real-world data collection. Extensive experimental evaluations show that our method surpasses conventional regression, matching, and registration-based baselines while demonstrating strong generalizability to previously unseen intra-category CAD models. Ablation studies further confirm that the render-compare design significantly enhances sim-to-real performance. Additionally, we demonstrate that our framework extends naturally to pose tracking, uncertainty estimation, and arbitrary tactile input. To facilitate future research, we will open-source our training and inference pipeline, along with training datasets and real-world evaluation data.

Limitations and Future Work. Our framework has two main limitations. First, although our pose tracking algorithm operates at 10 FPS during inference, the pose estimation procedure runs at less than 1 FPS, taking 1 to 2 seconds per pose estimate. This is significantly slower compared to methods such as [1]. This limitation could be alleviated by employing more advanced diffusion models, such as Flow-Matching [53]. While our approach requires over 500 steps of denoising during inference, Flow-Matching only requires tens of steps, making it several orders of magnitude faster. This suggests that it may be possible to accelerate our pipeline to over 25 FPS in the future. Second, our experiments have been conducted at the category level on a small set of objects from a single category. Future work will focus on validating our approach on a larger set of objects and assessing its generalization to novel objects from unseen categories.

Acknowledgments

We thank Jiyao Zhang, Tianhao Wu, and Jinzhou Li for their insightful discussions. This project was supported by the National Youth Talent Support Program (8200800081) and the National Natural Science Foundation of China (62376006).

References

[1] M. Bauza, A. Bronars, and A. Rodriguez. Tac2pose: Tactile object pose estimation from the first touch. *The International Journal of Robotics Research*, 42(13):1185–1209, 2023.

[2] B. Wen, W. Yang, J. Kautz, and S. Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17868–17879, 2024.

[3] M. Bauza, A. Bronars, Y. Hou, I. Taylor, N. Chavan-Dafle, and A. Rodriguez. Simple, a visuotactile method learned in simulation to precisely pick, localize, regrasp, and place objects. *Science Robotics*, 9(91):eadi8808, 2024.

[4] M. Bauza, O. Canal, and A. Rodriguez. Tactile mapping and localization from high-resolution tactile imprints. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3811–3817. IEEE, 2019.

[5] Y. Gao, S. Matsuoka, W. Wan, T. Kiyokawa, K. Koyama, and K. Harada. In-hand pose estimation using hand-mounted rgb cameras and visuotactile sensors. *IEEE Access*, 11:17218–17232, 2023.

[6] W. Gao and R. Tedrake. Filterreg: Robust and efficient probabilistic point-set registration using gaussian filter and twist parameterization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11095–11104, 2019.

[7] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[8] M. A. Schaeffer and A. M. Okamura. Methods for intelligent localization and mapping during haptic exploration. In *International Conference on Systems, Man and Cybernetics*. IEEE, 2003.

[9] C. Corcoran. Tracking object pose and shape during robot manipulation based on tactile information. In *International Conference on Robotics and Automation (ICRA)*, 2010.

[10] A. Petrovskaya and O. Khatib. Global localization of objects via touch. *IEEE Transactions on Robotics*, 27(3):569–585, 2011.

[11] M. Chalon, J. Reinecke, and M. Pfanne. Online in-hand object localization. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2977–2984. IEEE, 2013.

[12] J. Bimbo, S. Luo, K. Althoefer, and H. Liu. In-hand object pose estimation using covariance-based tactile to geometry matching. *IEEE Robotics and Automation Letters*, 2016.

[13] B. Saund, S. Chen, and R. Simmons. Touch based localization of parts for high precision manufacturing. In *International Conference on Robotics and Automation (ICRA)*. IEEE, 2017.

[14] S. Javdani, M. Klingensmith, J. A. Bagnell, N. S. Pollard, and S. S. Srinivasa. Efficient touch based localization through submodularity. In *International Conference on Robotics and Automation (ICRA)*. IEEE, 2013.

[15] Y. Chebotar, O. Kroemer, and J. Peters. Learning robot tactile sensing for object manipulation. In *International Conference on Intelligent Robots and Systems*. IEEE, 2014.

[16] M. C. Koval, N. S. Pollard, and S. S. Srinivasa. Pose estimation for planar contact manipulation with manifold particle filters. *The International Journal of Robotics Research*, 2015.

[17] M. C. Koval, M. Klingensmith, S. S. Srinivasa, N. S. Pollard, and M. Kaess. The manifold particle filter for state estimation on high-dimensional implicit manifolds. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4673–4680, May 2017.

[18] J. Bimbo, P. Kormushev, K. Althoefer, and H. Liu. Global estimation of an object's pose using tactile sensing. *Advanced Robotics*, 29(5):363–374, 2015.

[19] P. K. Allen, A. T. Miller, P. Y. Oh, and B. S. Leibowitz. Integration of vision, force and tactile sensing for grasping. *Int. J. Intelligent Machines*, 4:129–149, 1999.

[20] J. Ilonen, J. Bohg, and V. Kyrki. Three-dimensional object reconstruction of symmetric objects by fusing visual and tactile sensing. *The International Journal of Robotics Research*, 2014.

[21] P. Falco, S. Lu, A. Cirillo, C. Natale, S. Pirozzi, and D. Lee. Cross-modal visuo-tactile object recognition using robotic active exploration. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 5273–5280. IEEE, 2017.

[22] K.-T. Yu and A. Rodriguez. Realtime state estimation with tactile and visual sensing. application to planar manipulation. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7778–7785. IEEE, 2018.

[23] R. Platt, F. Permenter, and J. Pfeiffer. Using bayesian filtering to localize flexible materials during manipulation. *IEEE Transactions on Robotics*, 27(3):586–598, 2011.

[24] Z. Pezzementi, C. Reyda, and G. D. Hager. Object mapping, recognition, and localization from tactile geometry. In *International Conference on Robotics and Automation (ICRA)*, 2011.

[25] S. Luo, W. Mou, K. Althoefer, and H. Liu. Localizing the object contact through matching tactile features with visual map. *CoRR*, abs/1708.04441, 2017.

[26] N. Kuppuswamy, A. Castro, C. Phillips-Grafflin, A. Alspach, and R. Tedrake. Fast model-based contact patch and pose estimation for highly deformable dense-geometry tactile sensors. *IEEE Robotics and Automation Letters*, PP:1–1, 12 2019. doi:10.1109/LRA.2019.2961050.

[27] Y. Tu, J. Jiang, S. Li, N. Hendrich, M. Li, and J. Zhang. Posefusion: Robust object-in-hand pose estimation with selectlstm. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6839–6846. IEEE, 2023.

[28] H. Li, S. Dikhale, S. Iba, and N. Jamali. Vihope: Visuotactile in-hand object 6d pose estimation with shape completion. *IEEE Robotics and Automation Letters*, 2023.

[29] E. Donlon, S. Dong, M. Liu, J. Li, E. Adelson, and A. Rodriguez. Gelslim: A high-resolution, compact, robust, and calibrated tactile-sensing finger. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[30] F. R. Hogan, M. Bauzá, O. Canal, E. Donlon, and A. Rodriguez. Tactile regrasp: Grasp adjustments via simulated tactile transformations. *CoRR*, abs/1803.01940, 2018.

[31] S. Wang, J. Wu, X. Sun, W. Yuan, W. T. Freeman, J. B. Tenenbaum, and E. H. Adelson. 3D Shape Perception from Monocular Vision, Touch, and Shape Priors. *ArXiv e-prints*, Aug. 2018.

[32] S. Tian, F. Ebert, D. Jayaraman, M. Mudigonda, C. Finn, R. Calandra, and S. Levine. Manipulation by feel: Touch-based control with deep predictive models. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 818–824. IEEE, 2019.

[33] S. Yang, W. D. Kim, H. Park, S. Min, H. Han, and J. Kim. In-hand object classification and pose estimation with sim-to-real tactile transfer for robotic manipulation. *IEEE Robotics and Automation Letters*, 9(1):659–666, 2023.

[34] M. B. Villalonga, A. Rodriguez, B. Lim, E. Valls, and T. Sechopoulos. Tactile object pose estimation from the first touch with geometric contact rendering. In *Conference on Robot Learning*, pages 1015–1029. PMLR, 2021.

[35] A. Hyvärinen and P. Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

[36] P. Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

[37] Y. Song, S. Garg, J. Shi, and S. Ermon. Sliced score matching: A scalable approach to density and score estimation. In *Uncertainty in Artificial Intelligence*, pages 574–584. PMLR, 2020.

[38] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

[39] Y. Song and S. Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

[40] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[41] R. Cai, G. Yang, H. Averbuch-Elor, Z. Hao, S. Belongie, N. Snavely, and B. Hariharan. Learning gradient fields for shape generation. In *European Conference on Computer Vision*, pages 364–381. Springer, 2020.

[42] S. Luo and W. Hu. Score-based point cloud denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4583–4592, 2021.

[43] R. Shao, Z. Zheng, H. Zhang, J. Sun, and Y. Liu. Diffustereo: High quality human reconstruction via diffusion-based stereo using sparse cameras. *arXiv preprint arXiv:2207.08000*, 2022.

[44] H. Ci, M. Wu, W. Zhu, X. Ma, H. Dong, F. Zhong, and Y. Wang. Gfpose: Learning 3d human pose prior with gradient fields. *arXiv preprint arXiv:2212.08641*, 2022.

[45] J. Zhang, M. Wu, and H. Dong. Generative category-level object pose estimation via diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024.

[46] W. Chen, X. Jia, H. J. Chang, J. Duan, L. Shen, and A. Leonardis. Fs-net: Fast shape-based network for category-level 6d object pose estimation with decoupled rotation mechanism. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1581–1590, 2021.

[47] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5745–5753, 2019.

[48] S. Laine, J. Hellsten, T. Karras, Y. Seol, J. Lehtinen, and T. Aila. Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics*, 39(6), 2020.

[49] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

[50] T. Salimans and J. Ho. Should EBMs model the energy or the score? In *Energy Based Models Workshop - ICLR*, 2021. URL https://openreview.net/forum?id=9AS-TF2jRNb.

[51] J. R. Dormand and P. J. Prince. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980.

[52] E. Corona, K. Kundu, and S. Fidler. Pose estimation for objects with rotational symmetry. In *International Conference on Intelligent Robots and Systems*. IEEE, 2018.

[53] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

A Synthesizing Tactile Pose Estimation Dataset

We generate a tactile pose estimation dataset under practical conditions, where the objects are securely grasped in hand. As illustrated in Fig. 2 (I), for each object O, we first select several grasping approaches based on the object's canonical frame, such as $X+$ or $Y-$, which indicate that the left sensor will face the object along the corresponding axis for grasping. The object is initially positioned at the center of the TCP (Tool Center Point) frame. To introduce variability, randomization is applied in two dimensions: (1) along the xy-plane perpendicular to the grasping approach direction, and (2) in the rotational direction around the grasping approach axis. Specifically, the position in the xy-plane is randomized within $[-b_m, b_m]$, where b_m represents the maximum bounding box of the object in the given xy-plane. The gripper subsequently approaches the object along the selected grasping direction, with an indentation randomized within $[0.2\text{mm}, 1\text{mm}]$. Once the grasp is performed, we render the FEM-simulated left and right RGB tactile images. Finally, we filter out data points where the contact region is smaller than 5% of the sensor's imaging area to ensure data quality. Notably, we observed that ColorJitter serves as an effective augmentation strategy during training across all methods presented in this paper, as demonstrated by the ablation studies in Sec. 4.2. Specifically, we utilized the PyTorch implementation of ColorJitter with brightness, contrast, saturation, and hue parameters set to 0.3, 0.3, 0.3, and 0.1, respectively.

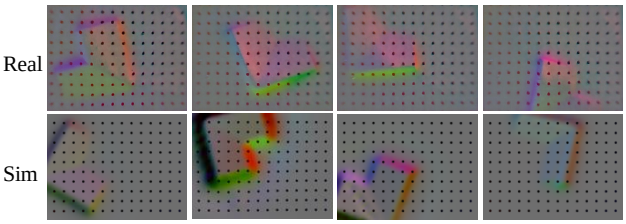


Figure 5: Simulation and real-world tactile images.

B Real-World Data Collection and Hardware Setup

Hardware Setup. We use the GelSlim 3.0 sensor for both simulation and real-world data collection, following the approach of [1]. This sensor provides high-resolution tactile readings in the form of RGB images. It consists of a deformable membrane that responds to contact, and an internal camera that captures the deformation. The sensor transmits tactile observations via ROS as 640×480 compressed images at a frequency of 90Hz. The object CAD model is 3D-printed and securely mounted on a table using a fixture. The sensor is attached to the end-effector of a Franka Panda robot, which randomly samples poses to establish contact with the fixed object model.

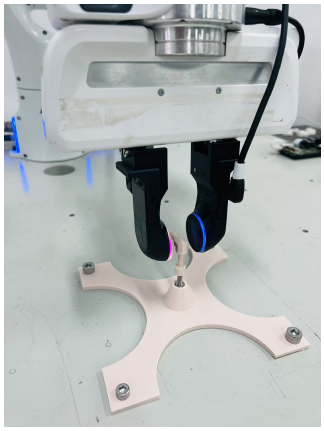


Figure 6: Hardware setup.

Real-world Data Collection. We collected labeled datasets of tactile observations for grasps performed on 30 objects. Each object was mounted in a known position and orientation. For each dataset, we collected pairs of RGB tactile images and their corresponding ground-truth object poses relative to the gripper center. A comparison of tactile imprints obtained from simulation and real-world experiments is shown in Fig. 5. The robotic system used for data collection includes the Franka Panda robot equipped with a GelSlim 3.0 tactile sensor on each finger. The hardware setup is illustrated in Fig. 6. **Data Collection Process.** The data collection procedure consists of the following steps:

- **Mounting the Object:** Each object is attached to a fixed platform using its threaded base. This ensures that the object remains stationary during grasping operations and prevents any slippage.
- **Calibrating Object Pose:** We initially replicate the real-world experimental setup within a simulation environment to estimate the object's pose relative to the gripper center. The pose is then refined by performing multiple grasps at different orientations and indentation

depths. We iteratively compare tactile images from real-world grasps with those from the simulation and adjust the simulated object pose until both sets of images align closely.

- **Data Collection:** For each object, predefined grasp axes and approach directions are established. We systematically gather tactile data by performing grasps along these directions, varying the gripper's x, y, and θ positions relative to the object. To simulate real-world uncertainties, we introduce in-plane rotational noise in each grasp. We also conduct grasps at three distinct indentation depths to capture variations in grasp quality and applied force.

C Baseline Implementation

FilterReg utilizes a partial point cloud extracted from the depth image produced by the tactile sensor. It is a probabilistic variant of the traditional ICP algorithm, aligning the tactile point cloud with the CAD model within a Gaussian Mixture Model (GMM) framework. FilterReg iteratively minimizes alignment errors to refine the estimated pose. For this method, we provide a rough estimate by introducing noise to the ground truth pose. We implement two variants of FilteReg: FilteReg(Global) matches the tactile point cloud with the complete point cloud derived from the CAD model. FilteReg(Partial) uses the oracle object pose and grasping depth to acquire partial point cloud from the CAD model, serving as an upper bound for FilterReg.

Vanilla Regression is a simple baseline where the pose is regressed directly from the tactile RGB image and the point cloud derived from the CAD model. The point cloud is encoded using PointNet, and the tactile image is processed through a combination of convolutional and Vision Transformer blocks, similar to our approach. Features from both modalities are fused and passed through a regression network to predict the pose.

Matching We reimplement Tac2Pose [1], as the original implementation has not been publicly released. For each object, we construct pose grids with a resolution of 2.5 mm and 6 degrees, resulting in approximately 8K to 17K grid points depending on the object geometry. The MoCo module [7] is trained for 30 epochs, ensuring convergence of the training loss for all objects. For fair comparison, the main differences from the original method are: (1) we omit training the image-to-image transfer module, thus making our method purely simulation-based; and (2) during real-world inference, we directly utilize the raw contact masks derived from depth measurements provided by the tactile sensor.

D Extension for Pose Tracking and Uncertainty Estimation

In this section, we provide a detailed description of the extensions to our framework for supporting pose tracking and uncertainty estimation. These additions are crucial for improving the robustness of object manipulation, particularly when dealing with dynamic environments and initial contact uncertainty.

D.1 Pose Tracking

In precise manipulation tasks, objects are often subject to motion, and it is essential to track their poses continuously over time to avoid unintended collisions with the environment. Our pose estimation framework can be naturally extended to pose tracking by modifying the prior distribution during the pre-filtering stage.

Tracking Prior Distribution. In the pose tracking scenario, the primary challenge is to incorporate temporal continuity from previously estimated poses. The tracking prior distribution is designed to reflect this temporal consistency, making it more informative compared to the prior used for the initial pose estimation.

Given the previously estimated in-hand pose $\hat{\boldsymbol{p}}_t$, we define the tracking prior distribution for the next frame as a Gaussian distribution:

$$\pi_{\text{track}}(\tilde{\boldsymbol{p}}_{t+1}) = \mathcal{N}(\tilde{\boldsymbol{p}}_{t+1}; \hat{\boldsymbol{p}}_t, \sigma_{track}^2 \mathbf{I}) \tag{9}$$

Here, $\sigma_{track} = 0.05$ is a hyperparameter that controls the spread of the distribution. This prior encourages the next pose to remain close to the previous estimate while allowing for some flexibility to account for small changes in pose due to motion or noise in the sensing.

Pose Tracking Procedure. Upon receiving a tactile observation T_{t+1} at frame $t+1$, we initialize the particle filter-based ODE solver (PF-ODE) in Eq. 6 with K candidates sampled from the tracking prior π_{track}. This approach leverages the previous pose estimate to more effectively predict the current pose.

Since pose tracking assumes continuity between consecutive frames, the tracking prior is typically closer to the high-density regions of the posterior distribution $p_{\text{data}}(\boldsymbol{p}|O, T)$, compared to the prior used for initial pose estimation. This allows the pose tracking process to be more efficient.

To further enhance the tracking efficiency, we set a smaller value for the time step parameter t_0 in the PF-ODE, specifically $t_0 = 0.1$, during tracking. This smaller time step allows the model to utilize more informative gradients, which leads to faster convergence during tracking compared to initial pose estimation.

Pose Selection. At the end of the tracking process, we select the pose candidate with the highest energy value as the final estimate. This candidate is the one that best matches the observed tactile feedback, considering both the prior distribution and the sensory information.

Efficiency of Pose Tracking. Compared to frame-by-frame pose estimation, pose tracking is significantly faster, with an inference rate of approximately 10 Hz, which is about 10 times faster than the initial pose estimation pipeline. This speed improvement comes from the reduced number of refinement steps required during tracking, as the previously estimated pose serves as a strong prior.

In practice, it is most beneficial to spend additional time accurately estimating the initial pose at the first contact frame, where the uncertainty is typically higher. Once the initial pose is accurately determined, subsequent frames can be processed efficiently with minimal refinement, allowing for real-time tracking.

D.2 Uncertainty Estimation

Uncertainty plays a crucial role in object manipulation, particularly when the object is grasped at uncertain contact points. Our framework quantifies uncertainty by calculating the variance of the refined pose candidates. The goal is to identify regions with lower uncertainty and guide the robot to re-grasp the object at those locations to improve pose estimation accuracy.

Relative Uncertainty Estimation. The uncertainty of the pose candidates $\{\hat{\boldsymbol{p}}_i\}_{i=1}^K$ is measured by computing their variance. This is done by first aggregating the candidate poses into a mean pose, denoted as $\hat{\boldsymbol{p}}_{\text{mean}}$, following the procedure described in GenPose [45].

Once the mean pose is computed, we define the uncertainty as the variance of the pose candidates. Specifically, the uncertainty S^2 is calculated as:

$$S^2 = \frac{1}{K} \sum_{i=1}^{K} d(\hat{\boldsymbol{p}}_i, \hat{\boldsymbol{p}}_{\text{mean}}, O) \tag{10}$$

Here, $d(\hat{\boldsymbol{p}}_i, \hat{\boldsymbol{p}}_{\text{mean}}, O)$ is a distance metric that quantifies the difference between the candidate pose $\hat{\boldsymbol{p}}_i$ and the mean pose $\hat{\boldsymbol{p}}_{\text{mean}}$ with respect to the object O. For non-symmetric objects, we use the Average Distance Difference (ADD) metric, while for symmetric objects, we use the ADD-S metric. Both of these metrics measure the average Euclidean distance between the model points under two different poses.

Grasp Comparison and Re-grasping. Given N different grasp candidates $\boldsymbol{g}_i i = 1^N$ that result in different tactile images $T_i i = 1^N$ of the object O, we compare the relative uncertainty of each grasp based on the computed variance S_i^2 of their corresponding pose candidates.

The relative uncertainty between two grasps $\boldsymbol{g}_i$ and $\boldsymbol{g}_j$ is determined as:

$$\boldsymbol{g}_i \succ \boldsymbol{g}_j \iff S_i^2 > S_j^2, \quad \boldsymbol{g}_{re} = \arg\min_i S_i^2 \tag{11}$$

In this context, the grasp with the lowest uncertainty, $\boldsymbol{g}_{re}$, is selected for re-grasping. This means that the robot will choose the grasp pose that minimizes uncertainty in the pose estimation, leading to more accurate and reliable manipulation.

Re-grasping for Improved Pose Estimation. The ability to re-grasp the object at regions of lower uncertainty is critical for improving pose estimation accuracy over time. This approach allows the robot to refine its understanding of the object's pose by re-grasping at more stable and distinguishable contact regions, reducing the effect of initial uncertainty and improving the overall robustness of the manipulation process.

E Additional Experimental Details

In this section, we provide detailed results and discussions for the experiments related to in-hand pose tracking, uncertainty estimation, and the handling of arbitrary contacts.

E.1 Pose Tracking and Uncertainty Estimation

We conduct extensive studies to evaluate the effectiveness of our method in two key aspects: in-hand pose tracking and relative uncertainty estimation.

In-hand Pose Tracking. We collect three real-world trajectories on three objects, each consisting of 50 time steps. The initial pose estimate for each tracking trajectory is obtained using our pose estimation method, and subsequent estimations are computed as described in Appendix D. As shown in Tab. 2, our method demonstrates robust performance, maintaining an error within 2mm across all three objects. Additionally, we observe a stable pose tracking frame rate of 10 Hz.

	Bear Housing(mm)	Rail(mm)	Deutsch Connector(mm)
Ours	1.2	1.8	1.5

Table 2: Pose Tracking Results.

Relative Uncertainty Estimation. We sample 10 sets of data points from the real-world test set of three objects, with each set containing data points from 10 different grasps. For each set, we apply the relative uncertainty estimation method (described in Appendix D) to select the Top-1, Top-3, and Top-5 grasps and compute the average pose error for each selection. As shown in Tab. 3, the pose errors for grasps selected using uncertainty estimation consistently outperform the baseline, where a grasp is randomly chosen from the 10 candidates. Furthermore, as the Top-K selection narrows, the average error decreases, demonstrating the effectiveness of the uncertainty estimation method.

E.2 Extending UniTac2Pose to Arbitrary Contacts

Our framework is designed to handle an arbitrary number of tactile contacts, thanks to its end-to-end training paradigm. To validate this, we compare three variants of the framework across six objects:

- *Double*: The original version of the framework, which uses two tactile images as input.
- *Single*: A variant that ablates the right sensor observation, using only the left tactile sensor for render-compare.

	Nut (mm)	Cotter (mm)	Cable Clip (mm)
Random Selection	2.8	2.9	9.9
Top-1 Confidence	1.5	0.6	1.5
Top-3 Confidence	2.1	0.8	4.7
Top-5 Confidence	2.6	1.0	7.7

Table 3: Grasp uncertainty estimation results. We compute the variance of estimated poses of a certain grasp generated by our model. Lower variance indicates lower uncertainty (higher confidence). We compute the mean ADD-S(mm) over top-k confident grasps. The ADD-S error of top-k grasps are lower than the mean error of random selected grasp, demonstrating the effectiveness of our grasp uncertainty estimation.

- *Arbitrary*: A variant where either the left or right tactile image is randomly masked during training and testing.

As shown in Tab. 4, *Double* consistently outperforms both *Single* and *Arbitrary*, as the latter two suffer from higher observation ambiguity. Although *Arbitrary* performs poorly on the *Stud* object, its overall performance is comparable to *Single*, demonstrating that our method can generalize well to scenarios where only a subset of tactile observations is available.

	Single (mm)	Double (mm)	Arbitrary (mm)
Cotter	1.8	1.5	2.5
Hose	2.9	1.2	2.4
Hydraulic	2.3	2.4	2.5
Round Nut	2.9	2.4	3.0
Rail	1.3	1.5	1.2
Stud	1.6	2.2	8.6

Table 4: Evaluation with different contact settings.

F Ablations on three stages.

We conduct ablation studies on the three stages of our method: pre-filtering, refinement, and post-ranking. We also compare refining only the top candidate pose. As shown in Tab. 5, our full method surpasses all variants, demonstrating the effectiveness of each stage.

	Ours	w/o pre-filter	w/o refine	w/o post-rank	refine top-1
Bear Housing	**2.4**	2.8	13.1	3.1	3.7
Cable Holder	**2.4**	2.7	12.0	3.3	4.2
Hose	**1.5**	1.6	16.0	1.7	1.6

Table 5: Ablation on three stages. Ours *w/o pre-filter* randomly samples 16 candidates from the prior, *w/o refine* selects top-1 candidate as the output, *w/o post-rank* reports the average ADD-S of 16 refined poses, and *refine top-1* means refining the top-1 candidate as the output.

G Choices of t.

For pose refinement, we train a model on the Hose object with the time parameter t uniformly sampled from the range $[0, 1]$. During inference, we initialize the RK45 solver with t values ranging from 0.4 to 1.0 in increments of 0.1 and evaluate the corresponding performance. As shown in Fig. 7, initialization with $t \geq 0.6$ results in consistent performance, whereas $t \leq 0.5$ leads to a notable drop

in accuracy. Based on this observation and inference efficiency, we choose $t = 0.6$ as the default setting. For pose selection, we vary t from 0.1 to 0.6 and evaluate performance across these values. As illustrated in Fig. 8, our pose selection method demonstrates robustness to the choice of t.

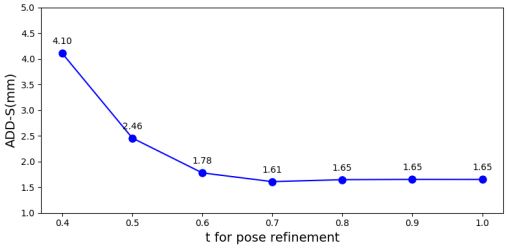


Figure 7: Evaluation of different initial t for pose refinement.

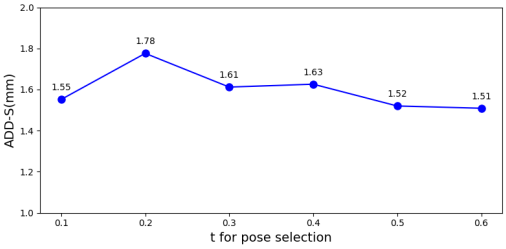


Figure 8: Evaluation of different t for pose selection.

H Training UniTac2Pose on objects from multiple categories.

We train a single model using data from three objects: Round Nut, Hose, and Rail. These objects were selected for their distinct shapes and differing symmetry properties—Round Nut exhibits half-turn symmetry, Rail has quarter-turn symmetry, and Hose is asymmetric. As shown in Tab. 6, the multi-object model successfully fits all three objects, achieving performance comparable to individual models trained separately for each object. This result suggests that our method is capable of learning across multiple object categories within a single model and holds promise for improved out-of-distribution (OOD) generalization as more diverse training data becomes available.

	Round Nut(mm)	Hose(mm)	Rail(mm)
Multi-object Model	2.3	1.5	1.7
Single-object Model	2.4	1.5	1.5

Table 6: Comparison of a model trained on three objects versus models trained on individual objects.

I The sim-to-real performance gap.

We report the evaluation performance in both simulation and the real world in Tab. 7. For most objects, the models achieve comparable performance in simulation. However, the Hook object is an exception, as its uneven surface results in severe partial observations and ambiguities in the contact images, ultimately degrading performance. The sim-to-real performance gap varies across objects, largely due to differences in contact surface geometry, object size, and symmetry. Overall, our method demonstrates a relatively small sim-to-real gap, owing to the use of extensive data augmentation and randomization during training.

	Bear Housing	Cable Holder	Cable Clip	Round Nut	Cotter	Hook	Hose	Hydraulic	Stud	Rail
Sim	1.4	0.8	1.0	1.1	1.3	3.5	0.9	1.3	1.3	1.4
Real	2.5	2.7	1.6	2.4	1.5	3.0	1.5	2.4	2.2	1.5
Δ	1.1	1.9	0.6	1.3	0.2	-0.5	0.6	1.1	0.9	0.1

Table 7: The sim-to-real performance gap.We report ADD-S (mm) and ADD (mm) errors for symmetric and non-symmetric objects respectively. Lower ADD/ADD-S error implies better performance.