ManipulationRAG: Retrieval-Augmented Fine-Grained Manipulation of Object Functional Parts

Xiaoyun Chang Zijun Zhang Jie Chen Xiaopeng Wen
Xiangbo Lin Xiaohong Ma Yi Sun*
School of Information and Communication Engineering
Dalian University of Technology

Abstract

Modeling fine-grained human manipulation of object functional parts is essential for achieving human-like dexterous manipulation in virtual reality and robotics. Despite recent advances, data-driven generation methods remain largely constrained by patterns learned from limited training data, rendering them prone to hallucinating task-inconsistent manipulations when encountering samples beyond the training distribution. To address this challenge, we propose ManipulationRAG, a framework that augments an internal generative model with externally retrieved, task-relevant manipulation knowledge. Specifically, we design a hand-centric manipulation taxonomy and construct a structured knowledge base that organizes task descriptions, objects, manipulation types, and corresponding hand pose sequences, supported by a retrieval mechanism for accurate knowledge retrieval. The retrieved knowledge serves as an external manipulation-specific prior, guiding the diffusion model toward task-consistent and physically plausible manipulation synthesis. Extensive experiments demonstrate that our method outperforms existing methods and generalizes robustly to object manipulation beyond the original diffusion model's training domain. We will release our dataset for further research at https://github.com/cgsmalcloud/ManipulationRAG.git.

1 Introduction

Fine-grained functional manipulation refers to object manipulation performed through precise finger control over small, task-relevant functional parts. As illustrated in Figure 1(a), clicking a button on a remote control or scrolling a mouse wheel require the fingertips to accurately contact tiny object parts, where small deviations in finger motion may lead to task failure. Such manipulations pose significant challenges for both modeling and generalization across diverse objects. Unlike grasping, which prioritizes grip stability without necessarily involving subsequent complex operations or object alterations,manipulation emphasizes a series of purposeful and skillful actions aimed at inducing targeted changes in the state, position, or configuration of an object's functional parts. This form of precise object manipulation holds critical importance for simulating fine hand actions in virtual reality [1] and for advancing dexterous manipulation in robotics [2].

To achieve manipulation synthesis, numerous methods have adopted data-driven paradigms, training models on large-scale annotated datasets with generative backbones based on variational autoencoders (VAEs) [3, 4], diffusion models [5–10], and large language models (LLMs) [11, 12].However, the generalization capacity of these methods remains fundamentally constrained by the scale of available datasets—a limitation that becomes particularly pronounced in the context of fine-grained manipulations on small functional parts of objects which involve subsequent complex operations. Consequently, contemporary manipulation models, being tightly bound to their limited data they were

*Corresponding author. lslwf@dlut.edu.cn

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

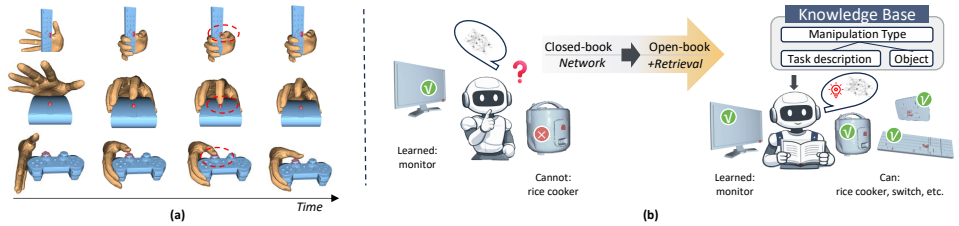


Figure 1: (a) Fine-grained functional manipulation examples. (b) Parameter-only models (closed-book) struggle with unseen objects, while retrieval (open-book) enables cross-category generalization.

trained on, struggle to generate credible fine-grained manipulations. In such scenarios, generative models are prone to "hallucination", potentially producing object manipulations that are semantically unrelated to the given task prompt.

To overcome these limitations, we draw inspiration from the observation that humans typically perform better in open-book exams than in closed-book ones. As illustrated in the left of Figure 1(b), when relying solely on data-driven learning, a model capable of operating the on/off switch of a monitor fails to generalize to the on/off switch of an electric rice cooker—an instance outside the training dataset. The internal knowledge embedded within a data-driven model alone is often insufficient, whereas access to external references enables the consultation of supplementary information to address previously unseen problems. Motivated by this insight, we explore Retrieval-Augmented Generation (RAG) [13] to ground the outputs of an internal generative model using external knowledge. Specifically, we propose ManipulationRAG, a method that dynamically retrieves manipulation-relevant knowledge based on a given task prompt and leverages this retrieved information as conditional guidance to steer the object manipulation generation process.

Under this retrieval-based framework, our method addresses two key challenges: i) obtaining effective and reliable manipulation priors, and ii) enabling the model to utilize them efficiently. To tackle the first challenge, we observe that despite the wide variety of objects, the manipulation types associated with their functional parts remain inherently limited. Motivated by this insight, we characterize human manipulation behaviors and formalize a taxonomy of manipulation action categories, which enables cross-object transfer of manipulation actions between functionally similar parts through retrieval of shared manipulation types. For the second challenge, we propose a retrieval-augmented diffusion model for fine-grained object manipulation. This model effectively leverages retrieved manipulation knowledge to generate more accurate and physically plausible manipulation actions on both familiar and novel objects. As illustrated on the right side of Figure 1, the skill of manipulating the on/off switch of a monitor generalizes to operating other objects with on/off switches—such as an electric rice cooker that are absent from the training dataset.

Our contributions are summarized as follows:

- **ManipulationRAG**: A retrieval-augmented diffusion model for fine-grained object manipulation that grounds generative outputs in external manipulation-specific knowledge for precise, context-aware synthesis.
- **Structured knowledge base & Retrieval**: A hand-centric manipulation taxonomy and a structured knowledge base organizing task descriptions, objects, manipulation types, and hand pose sequences, along with a retrieval mechanism for accurate knowledge retrieval.
- **Retrieval-augmented diffusion model**: A retrieval-augmented diffusion model that adaptively incorporates retrieved manipulation priors to enable robust, physically plausible, and generalizable fine-grained manipulation synthesis.

2 Related Work

2.1 Hand-Object Manipulation Synthesis

Modeling hand-object manipulation remains challenging due to diverse manipulation goals and complex contact patterns. Existing methods can be broadly divided into two categories. The first targets object grasping and lifting, typically through reinforcement learning [14, 15] or data-driven

motion generation [16, 17]. The second focuses on functional manipulation, such as tool use and task-specific interactions, often relying on conditional generative models, including VAEs [18] and diffusion models [19]. Recent language-conditioned methods further improve controllability by incorporating language guidance. Specifically, TaMF [20] and Text2HOI [5] both adopt coarse-to-fine generation strategies, where TaMF refines initially generated coarse hand motions, while Text2HOI first predicts contact heatmaps to guide motion synthesis and then further refines the generated motion. LatentHOI [9] improves generalization by encoding hand manipulation into a latent space with GraspVAE and decoding it using conditional diffusion. HOIGPT [12] and OpenHOI [11] incorporate large language models: the former directly generates hand motions, while the latter predicts affordances and decomposes complex tasks with multiple diffusion models. MoEG-HOI [8] further introduces Mixture-of-Experts into diffusion model for manipulation generation. Despite these advances, existing methods mainly focus on global hand-object manipulation. In contrast, we focus on fine-grained functional manipulation, such as clicking a button with a finger, which involves small contact regions and poses challenges for generalization to unseen objects. To address this, we propose a retrieval-based generation framework built on a structured manipulation knowledge base.

2.2 Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) was originally introduced for large language models to incorporate external knowledge into generation [13], and has since been extended to vision and generation tasks, including image classification [21, 22], object detection [23, 24], and human motion generation [25–27]. In human motion generation, retrieval-based methods such as ReMoDiffuse [25], ReMoGPT [26], and MoRAG [27] use text or motion priors to improve motion quality, semantic consistency, and diversity. Recently, RAG has also started to be explored for static grasp synthesis. For example, RAGG [28] combines discrete task intent representations with object BPS [29] features to retrieve relevant grasp examples from a prior database, and uses them to guide grasp synthesis. G-DexGrasp [30] further introduces part-aware prior retrieval, where affordance types and contact parts on the object are inferred from task instructions, and corresponding local grasp priors are retrieved to assist a CVAE [18] model and a post-optimization step for final grasp generation. However, these methods may rely on grasp priors to ensure stability, which are insufficient for fine-grained functional manipulation where success depends not only on contact locations but also on dynamic finger motion. To this end, we build a precise manipulation knowledge base to retrieve relevant manipulation and incorporate it into a diffusion model for robust and generalizable fine-grained manipulation synthesis.

3 Method

Given a task description t and an object point cloud o containing a functional part p, this paper aims to synthesize fine-grained manipulations targeting the specified part. To address this problem, we propose a novel framework that leverages retrieved hand pose sequences as manipulation references, allowing the model to effectively exploit prior manipulation knowledge and perform precise operations across diverse object instances. Figure 2 illustrates the overall architecture of the proposed framework. First, we construct a manipulation knowledge database to support fine-grained functional manipulation (Section 3.1). Based on this database, we retrieve the most relevant manipulation prior conditioned on the input object and task description (Section 3.2). The retrieved prior is then incorporated into a diffusion model to steer the synthesis of fine-grained manipulations (Section 3.3). We provide the implementation details in Section 3.4.

3.1 Knowledge Base for Fine-Grained Functional Manipulation

Fine-grained manipulation requires precise finger motions on small, task-relevant object parts, making the collection of demonstrations particularly difficult and costly. As a result, manipulation datasets are often limited in scale, which makes it challenging for purely data-driven generative models to learn reliable manipulation behaviors, especially when generalizing to previously unseen objects. This compels us to seek a knowledge-driven approach that simplifies annotation while enabling manipulation transfer across diverse object categories. While the diversity of object instances is virtually unbounded, their functional parts are inherently designed to accommodate human hand manipulation habits. Consequently, functional components across different objects, e.g., switches or buttons, exhibit recurring manipulation patterns, whose variability is substantially limited compared

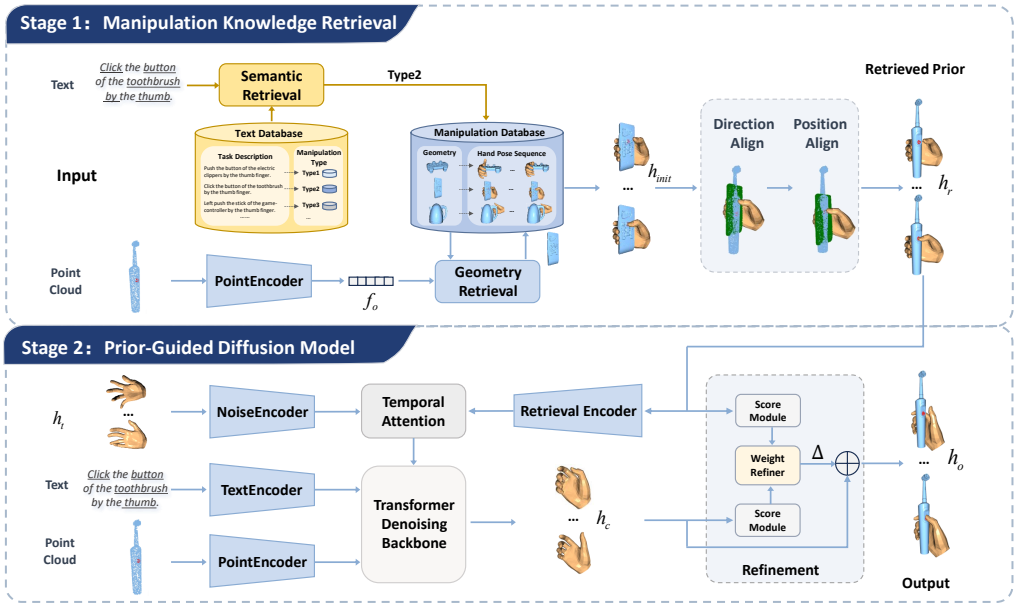


Figure 2: Overview of ManipulationRAG. Given an input object and a task description, we retrieve the most relevant hand pose sequence from a structured knowledge base, align it with the target object, and use the retrieved prior to adaptively guide the diffusion model.

to the vast spectrum of object geometries. This observation motivates us to pursue a more efficient pathway to cross-object manipulation: structuring hand-centric manipulations according to their underlying operation patterns rather than object-specific semantics.

Early work by Feix et al. [31] define 33 grasp types in a static manner, which are incompatible with subsequent manipulations. More recently, Wu et al. [32] group functional grasps for manipulation into 9 grasp types, yet these remain static as well. Inspired by [32], this paper extends the approach into a three-level taxonomy for dynamic functional manipulation, as illustrated in Figure 3. We organize manipulation actions into reusable primitives that can generalize across objects, and design a hand-centric manipulation taxonomy which captures transferable manipulation patterns. This structured knowledge provides general manipulation priors to complement data-driven learning model, enabling the model to perform reliable manipulations across both seen and unseen objects. The proposed taxonomy for dynamic functional manipulation comprises three levels. At the first level, manipulations are categorized according to the opposition relationship, including i) palm opposition, the object is stabilized by opposition between the palm and the fingers during manipulation, ii) pad opposition, where the object is held through opposition between two or more finger pads, and iii) no opposition, where the object is not maintained by opposing finger contacts. At the second level, each opposition category is further subdivided according to the active fingers involved in the manipulation, with the hand partitioned into multiple finger groups. For instance, in the palm opposition examples shown in the left of Figure 3, manipulating the camera (column 2) involves 2-finger group: the thumb, which clicks the button, and the remaining four fingers, which grasp the camera body. At the third level, manipulation types are further distinguished by finger motions, as thumb clicking and thumb pressing from the same finger group may still exhibit distinct dynamic actions. Finally, based on this three-level taxonomy, we define 13 manipulation types that capture transferable patterns in fine-grained functional manipulation.

We further construct a structured knowledge base to support fine-grained functional manipulation under this taxonomy. The knowledge base consists of two components: a task text database and a hand pose sequence database. The task text database contains 41 textual descriptions of object manipulation tasks. These tasks are performed on objects with one or more functional parts, such as flashlight with a single functional button and mouse with multiple functional buttons, as illustrated in Figure 4. Due to the diversity of object categories and instances, these task descriptions correspond to a wide range of hand motion sequences. Each description specifies an action in which a particular finger manipulates a functional part of a target object category, e.g., *'click the button of the electric*

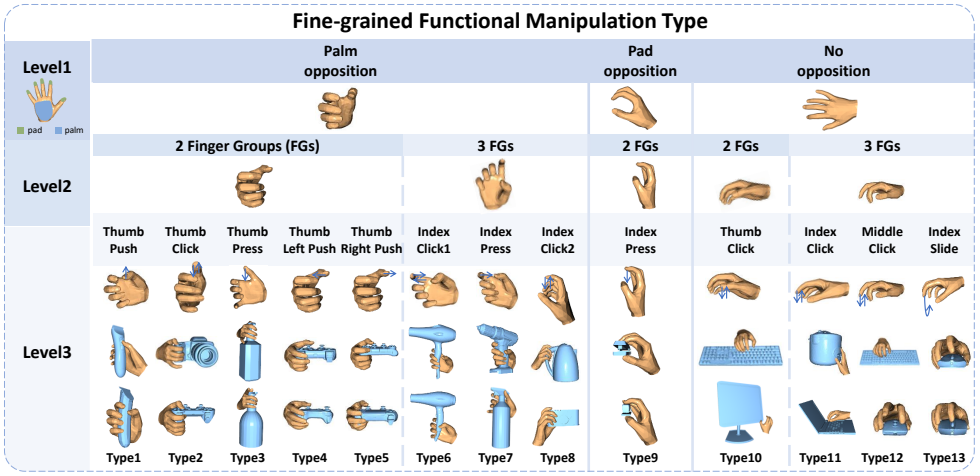


Figure 3: A three-level taxonomy for fine-grained functional manipulation. The direction of the arrows represents the direction of finger movement.

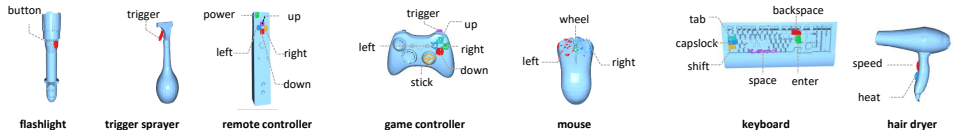


Figure 4: Visualization of object instances, where functional parts are indicated by distinct colors.

toothbrush by the thumb'. We categorize all these task descriptions according to our predefined 13 manipulation types. For example, the two task descriptions *'click the heat button of the hair dryer with the index finger'* and *'click the speed button of the hair dryer with the index finger'* are both assigned to Type 6, as they share the same palm-opposition, involve 3-finger groups, and use the index finger for clicking. The task-to-type associations are stored as natural-language sentences in a text document, enabling efficient lookup of the corresponding manipulation type given a task description during subsequent retrieval.

Given that the same manipulation type can involve multiple object instances and that geometric variations across these instances may induce local differences in hand poses, we further construct multiple representative objects and annotate hand pose sequences for each type. For example, Type12 involves keyboards and mice. We cluster all keyboard and mouse instances in the existing dataset and identify 16 representative object instances, each annotated with hand pose sequences, yielding 31 sequences in total. This design can retrieve similar objects and their corresponding hand pose sequences within a manipulation type. In total, across 13 manipulation types, we obtain 123 representative object instances from 17 object categories, together with 257 annotated hand pose sequences. Each object instance is encoded with geometric features extracted by a pre-trained point cloud encoder, PointBERT [33], while each hand pose sequence is represented by MANO [34] parameters, denoted as $h \in \mathbb{R}^{l \times d}$, where $l = 64$ is the temporal length and $d = 99$ includes 3 dimensions for hand translation and the remaining dimensions for the 6D hand pose representation.

By organizing manipulation knowledge into a structured database with 13 manipulation types, we summarize fine-grained manipulation priors that can serve as external references for purely parametric generative models and guide them toward more plausible manipulation generation. For unseen objects, the database can still identify an appropriate manipulation type aligned with the task description and provide a similar representative object manipulation as a reference, thereby improving the generalization ability of the generative model.

3.2 Manipulation Knowledge Retrieval

In this stage, we aim to retrieve a reference hand pose sequence h_r for subsequent generation, conditioned on the task description t and the target object o. Leveraging the structured knowledge base, we retrieve the most relevant hand pose sequence and further align it with the target object.

As illustrated in Figure 2, our retrieval framework proceeds in two steps: semantic retrieval and geometric retrieval. First, given a task description t, we employ a LightRAG-based module [35] to query the task text database and infer the corresponding manipulation type m. Second, conditioned on the retrieval manipulation type, we perform geometric retrieval over the manipulation database. Specifically, the point cloud of the target object o is encoded by PointBERT [33] to obtain its geometric feature f_o. We compute the L2 distance between f_o and the geometric features of all object instances associated with manipulation type m in the retrieval database, and identify the nearest object instance. The hand pose sequence associated with this instance is then retrieved as the initial manipulation reference for the target object, denoted as h_{init}.

However, the retrieved hand pose sequence h_{init} may not align well with the target functional part because it comes from the nearest object in the database. To fix this, we use an alignment module that transforms the retrieved manipulation to the target's. First, we align the direction using the surface normals, then the spatial positions using a position transformation. Finally, we obtain the aligned hand pose sequence h_r.

3.3 Retrieval-Augmented Diffusion Model for Functional Manipulation

In this stage, we investigate how the retrieved reference manipulation h_r can be effectively leveraged to improve the capability of diffusion models for fine-grained manipulation synthesis. We design a prior-guided diffusion model that adaptively incorporates h_r into the denoising process, enabling the synthesis of more accurate manipulations.

Given a task description t, an object point cloud o, and a retrieved reference h_r, our denoising network progressively transforms an initial noisy input h_t into the target manipulation sequence h_o. An overview of the network architecture is shown in Figure 2. The denoising network is composed of a coarse generation module and a refinement module. In the coarse generation module, the input task description is encoded using a pretrained CLIP text encoder [36], and the object point cloud is processed by a pretrained PointBERT encoder [33]. For the noisy input, we incorporate retrieved prior via a temporal attention-based fusion mechanism, enabling effective integration of retrieval information into the generation process. The resulting features from all modalities are aggregated as token inputs into a Transformer encoder [37], producing a coarse prediction h_c.

The refinement module further improves the coarse prediction. Previous generation model produces rough action outputs after initially understanding the retrieved manipulation knowledge. During the fine-tuning stage, the coarse outputs from the model and the externally retrieved knowledge enter a "contention" phase for dominance. The fine-tuned model comprehensively evaluates the reliability of the two information sources and ultimately produces h_o as a weighted fusion of the internal network knowledge h_c and the external knowledge h_r. The criteria for the reliability of h_r or h_c is based on a contact score S_{contact} for assessing contact quality, a penetration score S_{pen} for measuring collision, and a part accuracy score S_{part} for evaluating whether the text-specified finger contacts the target part. The overall score is computed as:

$$S = \frac{w_c S_{\text{contact}} + w_p S_{\text{pen}} + w_a S_{\text{part}}}{w_c + w_p + w_a}, \tag{1}$$

where w_c, w_a, and w_p are hyperparameters. Specifically, $S_{\text{contact}} = \exp(-\frac{1}{k}\sum_{i=1}^{k} d_i)$ measures the tightness of hand-object contact based on the average distance d_i of the closest k hand-object point pairs, $S_{\text{pen}} = \exp(-d_{\text{pen}})$penalizes hand-object interpenetration quantified by d_{pen}, and $S_{\text{part}} = \mathbb{I}(\text{contact})$ evaluates whether the finger specified in the description is in contact with the target region. The reliability of h_c and h_r determines their fusion weight, enabling the model to fully leverage both internal network knowledge and external retrieved knowledge, thereby achieving better manipulation generation capability. In the experimental section, we will demonstrate the performance gap that emerges when the model is trained exclusively on data, without access to external retrieval knowledge.

Losses. To train the denoising network, we supervise the refined output h_o with the ground truth using the following objective:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{MANO}} + \lambda_2 \mathcal{L}_{\text{joints}} + \lambda_3 \mathcal{L}_{\text{vertices}} + \lambda_4 \mathcal{L}_{\text{edge}} + \lambda_5 \mathcal{L}_{\text{distance}}. \tag{2}$$

Here, $\mathcal{L}_{\text{MANO}} = \|h_o - \hat{h}_o\|_2$ denotes the reconstruction loss between the predicted MANO parameters h_o and the ground-truth $\hat{h}_o$. The remaining terms measure geometric discrepancies between the

reconstructed motion and the ground truth, including joint positions $\mathcal{L}_{\text{joints}}$, hand vertices $\mathcal{L}_{\text{vertices}}$, edge lengths $\mathcal{L}_{\text{edge}}$, and hand-object distances $\mathcal{L}_{\text{distance}}$, following [20]. λ_1 to λ_5 are hyperparameters.

3.4 Implement details

For semantic retrieval, we adopt Qwen3-32B [38] as the backbone of LightRAG [35]. For the diffusion model, the denoising network is trained on a single NVIDIA RTX 4090 GPU. We use the Adam optimizer [39] with a learning rate of 1×10^{-4} and a batch size of 16, training the diffusion model for 500 epochs. The learning rate is decayed by a factor of 0.1 at epochs 200 and 300.

4 Experiments

In this section, we introduce the datasets and evaluation metrics in Section 4.1. This is followed by comparisons with state-of-the-art methods to validate the proposed method's generalization. Finally, ablation studies are conducted to analyze the effectiveness of the introduced modules in Section 4.3.

4.1 Datasets and Evaluation Metrics

Datasets. To address the limited availability of datasets for fine-grained functional manipulation scenarios, we construct a new dataset to evaluate our method. We collect 582 functional objects across 20 categories, including 15 categories with a single functional part (e.g., cameras, electric toothbrushes, trigger sprayers, and flashlights) and 5 categories with multiple functional parts (e.g., remote controls, game controllers, and hair dryers). All objects in our dataset are sourced from publicly available resources, including three public datasets [40–42] and several public 3D model websites [43–47]. Each object is manually annotated with functional parts, and its hand pose sequences are further annotated following DexFuncGrasp [48]. In total, we collect 1,879 sequences across 41 task descriptions. Each sequence is annotated with task descriptions, and corresponding MANO-based [34] hand pose sequences. The training set contains 1,283 sequences covering 17 object categories, while the test set contains 379 sequences and evaluates generalization to unseen objects within a category, unseen functional parts within a category, and unseen categories.

Evaluation Metrics. We adopt Contact Ratio and Solid Intersection Volume from [20] to evaluate physical plausibility, and use Diversity from [9] to measure the variability of manipulations. In addition, we introduce Part Accuracy to assess the alignment between generated manipulations and task descriptions. Specifically, Contact Ratio is defined as the proportion of frames in which the hand and object are in contact, where contact is determined by a surface distance below 5 mm. Solid Intersection Volume quantifies hand-object interpenetration by voxelizing the object mesh into a 100^3 grid and counting the voxels that fall inside the hand volume. Part Accuracy measures the proportion of samples in which task-specified fingers correctly contact the corresponding functional object parts. Diversity is computed as the average pairwise L2 distance across all generated samples.

4.2 Comparisons with Other Methods

To validate the effectiveness of the proposed method, we compare it with the most related state-of-the-art methods. These methods take text and objects as input and generate manipulation sequences based on diffusion models. Among them, TaMF [20] and Text2HOI [5] adopt coarse-to-fine multi-stage generation strategies. LatentHOI [9] incorporates a VAE into the diffusion framework, leveraging an implicit diffusion to improve generalization. To ensure a fair comparison, we retrain all baselines on our constructed dataset using their publicly available code.

As shown in Table 1, our method achieves the best Solid Intersection Volume, Part Accuracy, and Diversity, while maintaining a comparable Contact Ratio to existing approaches. These results demonstrate its strong physical plausibility and generation diversity. Notably, the superior Part Accuracy further shows its ability to identify semantically consistent manipulation regions on unseen objects. This improvement is attributed to the use of additional reference knowledge beyond parametric models, which enables more accurate identification of manipulation regions.

Additionally, Figure 5 shows qualitative comparisons with other methods using representative frames of generated manipulations. As shown, for the mouse, which is an unseen object from a category observed during training, our method accurately contacts the wheel with fewer collisions than other

Table 1: Quantitative comparison with other methods.

Method	Part Accuracy ↑	Contact Ratio ↑	Solid Intersection Volume ↓	Diversity ↑
GroundTruth	0.9947	0.9572	5.11	0.1099
TaMF [20]	0.1953	0.9384	22.85	0.1006
Text2HOI [5]	0.1953	**0.9844**	31.12	0.0799
LatentHOI [9]	0.2084	0.9596	28.73	0.0899
Ours	**0.9446**	0.9312	**8.04**	**0.1326**

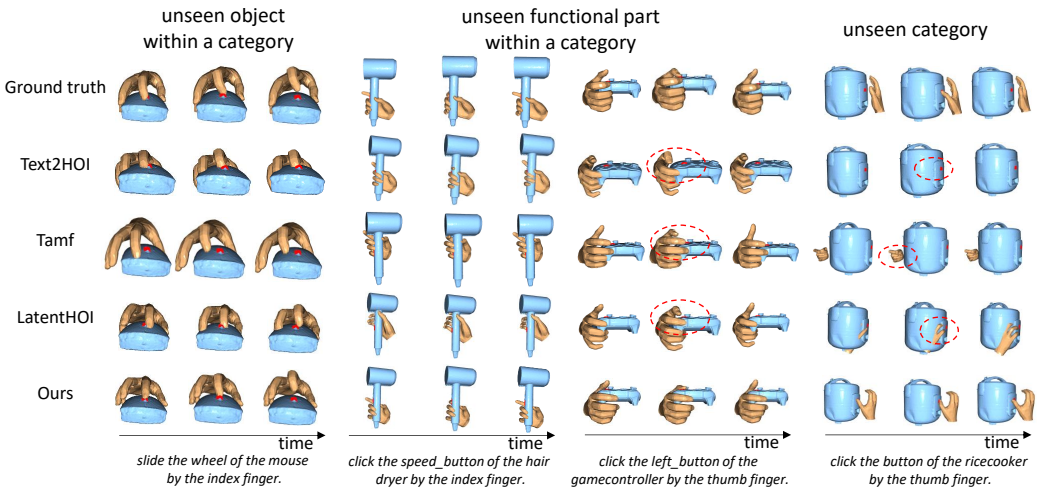


Figure 5: Qualitative results of comparison to other methods.

methods. Moreover, when the functional parts are unseen during training, prior methods may generate plausible hand poses but often fail to accurately interact with these novel parts. For example, for a game controller with an unseen left button, existing methods fail to establish accurate contact with the target, causing severe collisions or leaving the fingers far from it, whereas our method achieves fine-grained contact. For the rice cooker from an unseen category, other methods generate abnormal manipulations with severe collisions and fail to perform button clicking. In contrast, our method accurately localizes and clicks the power button, demonstrating robust generalization to unseen categories. More visualizations and videos are provided in the appendix and supplementary materials.

4.3 Ablation Study

Our method addresses two key challenges: i) how to obtain effective and reliable manipulation priors, and ii) how to enable the model to efficiently utilize these priors. Accordingly, the ablation studies are designed to verify whether these two challenges have been effectively addressed, with the retrieval and alignment modules targeting the first, and the prior-guided diffusion model targeting the second.

Retrieval Module. To assess the contribution of the retrieval module, we construct three ablation variants by removing: i) semantic retrieval, which uses the task instruction to infer manipulation type as a prior, ii) geometric retrieval, which retrieves a hand pose sequence from object with the most similar geometry as a prior, and iii) all retrieval modules. As shown in Table 2, removing geometric retrieval leads to a clear performance drop across most metrics, especially Part Accuracy. This is because the model loses structurally aligned priors from similar objects, making it harder to localize the target functional part. Removing semantic retrieval causes a smaller yet consistent decline: while geometric similarity still provides structural guidance, the lack of semantic cues weakens the alignment between the task description and the retrieved hand pose sequence, leading to less appropriate manipulation behaviors. When all retrieval modules are removed, performance drops substantially, as the model rely solely on its internal parameters without external priors, limiting its generalization. Qualitative results in Figure 6 further show that all variants struggle to varying degrees when pressing the button on a power drill. These results demonstrate that the retrieval module provides reliable priors that improve both accuracy and generalization in manipulation synthesis.

Table 2: Ablation study on different retrieval components.

Method	Part Accuracy ↑	Contact Ratio ↑	Solid Intersection Volume ↓	Diversity ↑
Ours	**0.9446**	0.9312	**8.04**	0.1326
w/o geometry	0.3483	**0.9429**	16.73	0.1065
w/o semantic	0.8522	0.9158	12.70	**0.1900**
w/o all retrieval	0.3193	0.9356	14.06	0.1034
w/o alignment	0.3562	0.9130	14.48	0.1111

Table 3: Ablation study on different components of the prior-guided diffusion model.

Method	Part Accuracy ↑	Contact Ratio ↑	Solid Intersection Volume ↓	Diversity ↑
w/o guidance	0.3193	0.9356	14.06	0.1034
coarse	0.5224	**0.9403**	9.88	0.1101
refinement	**0.9446**	0.9312	**8.04**	**0.1326**

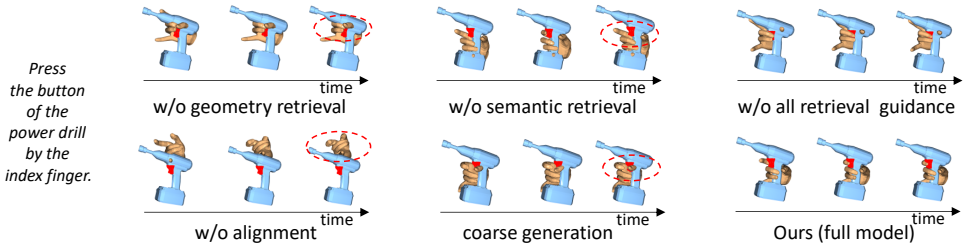


Figure 6: Qualitative ablation comparison across different components.

Alignment Module. To assess the alignment module, we ablate it from the full model. As shown in Table 2, the model performance decreases across all metrics, with the most notable drop observed in Part Accuracy. Qualitative results are presented in Figure 6. Without the alignment module, the generated manipulation fails to accurately target the button on a power drill from an unseen object category. This suggests that the alignment module is crucial for adapting retrieved manipulation priors to the functional regions of novel objects, thereby enabling more effective prior utilization.

The Components of Prior-guide Diffusion Model. To assess the coarse and refinement modules, we conduct an ablation study with three variants: i) w/o guidance, where the retrieved guidance is removed, ii) coarse generation, and iii) the full model with refinement. As shown in Table 3, introducing coarse guidance already yields consistent improvements over the variant without guidance, suggesting that the retrieved prior provides informative signals for generation. Applying refinement further improves most metrics, especially in Part Accuracy. The qualitative results in Figure 6 show that coarse guidance and refinement enable our method to accurately press the button of the power drill from an unseen category. These results demonstrate that the retrieval-guided module effectively exploits retrieved priors, thereby enhancing both physical plausibility and generalization.

5 Limitations and Future Work

Functional object parts, such as buttons on remote controls, are often very small in scale, posing significant challenges for accurate segmentation using existing 3D segmentation networks. To circumvent this issue, the present work assumes the availability of functional part annotations. While the constructed database encompasses a diverse range of representative object instances, it may still fall short when generalizing to novel objects whose local geometric structures or overall shapes deviate substantially from those in the database. In such scenarios, the generated hand pose sequences can retain plausible manipulation types, yet may exhibit reduced accuracy in local contact details. Future work will expand the database to enhance both coverage and generalization capability.

6 Conclusion

In this paper, we propose ManipulationRAG, a retrieval-augmented diffusion framework for fine-grained manipulation synthesis. Specifically, we design a hand-centric manipulation taxonomy and construct a structured knowledge base that organizes task descriptions, objects, manipulation types, and hand pose sequences. Based on this knowledge base, we retrieve task-relevant manipulation priors and incorporate them into the diffusion model to guide accurate and physically plausible manipulation synthesis. Extensive experiments demonstrate that our method significantly improves the robustness and generalization of fine-grained manipulation synthesis across diverse objects.

Acknowledgments and Disclosure of Funding

This work was supported by the National Natural Science Foundation of China under Grant No. 62373075.

References

[1] H. Jiang, D. Weng, Z. Song, X. Dongye, and Z. Zhang, "Dexhand: dexterous hand manipulation motion synthesis for virtual reality," *Virtual Reality*, vol. 27, no. 3, pp. 2341–2356, 2023.

[2] K. Li, P. Li, T. Liu, Y. Li, and S. Huang, "Maniptrans: Efficient dexterous bimanual manipulation transfer via residual learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6991–7003, 2025.

[3] A. Ghosh, R. Dabral, V. Golyanik, C. Theobalt, and P. Slusallek, "Imos: intent-driven full-body motion synthesis for human-object interactions," in *Computer Graphics Forum*, vol. 42, pp. 1–12, Wiley Online Library, 2023.

[4] J. Zheng, Q. Zheng, L. Fang, Y. Liu, and L. Yi, "Cams: Canonicalized manipulation spaces for category-level functional hand-object manipulation synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 585–594, 2023.

[5] J. Cha, J. Kim, J. S. Yoon, and S. Baek, "Text2hoi: Text-guided 3d motion generation for hand-object interaction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1577–1585, 2024.

[6] S. Christen, S. Hampali, F. Sener, E. Remelli, T. Hodan, E. Sauser, S. Ma, and B. Tekin, "Diffh2o: Diffusion-based synthesis of hand-object interactions from textual descriptions," in *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024.

[7] J. Zhang, Y. Zhang, L. An, M. Li, H. Zhang, Z. Hu, and Y. Liu, "Manidext: Hand-object manipulation synthesis via continuous correspondence embeddings and residual-guided diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[8] H. Xu, Y. Xiao, C. Jiang, H. Kuang, K. Zhang, M. Du, and R. Wang, "Moeg-hoi: Mixture of expert groups for one-stage hand-object interaction motion generation with hand-finger-joint semantic guidance," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, pp. 11223–11231, 2026.

[9] M. Li, S. Christen, C. Wan, Y. Cai, R. Liao, L. Sigal, and S. Ma, "Latenthoi: On the generalizable hand object motion generation with latent hand diffusion.," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 17416–17425, 2025.

[10] Z. Wang, L. Liu, R. Liu, D. Guo, and M. Wang, "Structbihoi: Structured articulation modeling for long–horizon bimanual hand–object interaction generation," *arXiv preprint arXiv:2603.08390*, 2026.

[11] Z. Zhang, Y. Shi, L. Yang, S. Ni, Q. Ye, and J. Wang, "Openhoi: Open-world hand-object interaction synthesis with multimodal large language model," in *Advances in Neural Information Processing Systems*, vol. 38, pp. 166582–166612, 2025.

[12] M. Huang, F.-J. Chu, B. Tekin, K. J. Liang, H. Ma, W. Wang, X. Chen, P. Gleize, H. Xue, S. Lyu, *et al.*, "Hoigpt: Learning long-sequence hand-object interaction with language models," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 7136–7146, 2025.

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

[14] S. Christen, M. Kocabas, E. Aksan, J. Hwangbo, J. Song, and O. Hilliges, "D-grasp: Physically plausible dynamic grasp synthesis for hand-object interactions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20577–20586, 2022.

[15] H. Zhang, S. Christen, Z. Fan, O. Hilliges, and J. Song, "Graspxl: Generating grasping motions for diverse objects at scale," in *European Conference on Computer Vision*, pp. 386–403, Springer, 2024.

[16] S. Shimada, F. Mueller, J. Bednarik, B. Doosti, B. Bickel, D. Tang, V. Golyanik, J. Taylor, C. Theobalt, and T. Beeler, "Macs: Mass conditioned 3d hand and object motion synthesis," in *2024 International Conference on 3D Vision (3DV)*, pp. 1082–1091, 2024.

[17] M. Zhang, Y. Ye, T. Shiratori, and T. Komura, "Manipnet: neural manipulation synthesis with a hand-object spatial representation," *ACM Transactions on Graphics*, p. 1–14, Aug 2021.

[18] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," *Advances in neural information processing systems*, vol. 28, p. 3483–3491, 2015.

[19] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[20] X. Zhan, L. Yang, Y. Zhao, K. Mao, H. Xu, Z. Lin, K. Li, and C. Lu, "Oakink2: A dataset of bimanual hands-object manipulation in complex task completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 445–456, 2024.

[21] H. Liu, K. Son, J. Yang, C. Liu, J. Gao, Y. J. Lee, and C. Li, "Learning custozed visual models with retrieval-augmented knowledge," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15148–15158, 2023.

[22] A. Long, W. Yin, T. Ajanthan, V. Nguyen, P. Purkait, R. Garg, A. Blair, C. Shen, and A. Van den Hengel, "Retrieval augmented classification for long-tail visual recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6959–6969, 2022.

[23] J. Kim, E. Cho, S. Kim, and H. J. Kim, "Retrieval-augmented open-vocabulary object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17427–17436, 2024.

[24] J. Du, X. Wang, F. Hao, M. Yu, C. Chen, J. Wu, B. Wang, J. Xu, and P. Li, "Beyond single images: Retrieval self-augmented unsupervised camouflaged object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22131–22142, 2025.

[25] M. Zhang, X. Guo, L. Pan, Z. Cai, F. Hong, H. Li, L. Yang, and Z. Liu, "Remodiffuse: Retrieval-augmented motion diffusion model," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 364–373, 2023.

[26] Q. Yu, M. Tanaka, and K. Fujiwara, "Remogpt: Part-level retrieval-augmented motion-language models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, pp. 9635–9643, 2025.

[27] S. S. Kalakonda, S. Maheshwari, and R. K. Sarvadevabhatla, "Morag-multi-fusion retrieval augmented generation for human motion," in *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 4564–4573, IEEE, 2025.

[28] Z. Tang, B. Zhu, Y. Hao, C.-W. Ngo, and R. Hong, "Ragg: Retrieval-augmented grasp generation model," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, pp. 7311–7319, 2025.

[29] S. Prokudin, C. Lassner, and J. Romero, "Efficient learning on point clouds with basis point sets," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4332–4341, 2019.

[30] J. Jian, X. Liu, Z. Chen, M. Li, J. Liu, and R. Hu, "G-dexgrasp: Generalizable dexterous grasping synthesis via part-aware prior retrieval and prior-assisted generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11447–11457, 2025.

[31] T. Feix, J. Romero, H.-B. Schmiedmayer, A. M. Dollar, and D. Kragic, "The grasp taxonomy of human grasp types," *IEEE Transactions on human-machine systems*, vol. 46, no. 1, pp. 66–77, 2015.

[32] R. Wu, T. Zhu, X. Lin, and Y. Sun, "Cross-category functional grasp transfer," *IEEE Robotics and Automation Letters*, vol. 9, no. 11, pp. 10652–10659, 2024.

[33] X. Yu, L. Tang, Y. Rao, T. Huang, J. Zhou, and J. Lu, "Point-bert: Pre-training 3d point cloud transformers with masked point modeling," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19313–19322, 2022.

[34] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands," *ACM Transactions on Graphics*, vol. 36, no. 6, pp. 1–17, 2017.

[35] Z. Guo, L. Xia, Y. Yu, T. Ao, and C. Huang, "Lightrag: Simple and fast retrieval-augmented generation," in *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 10746–10761, 2025.

[36] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*, pp. 8748–8763, 2021.

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

[38] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, *et al.*, "Qwen3 technical report," *arXiv preprint arXiv:2505.09388*, 2025.

[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[40] L. Yang, K. Li, X. Zhan, F. Wu, A. Xu, L. Liu, and C. Lu, "Oakink: A large-scale knowledge repository for understanding hand-object interaction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20953–20962, 2022.

[41] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang, *et al.*, "Sapien: A simulated part-based interactive environment," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11097–11107, 2020.

[42] T. Wu, J. Zhang, X. Fu, Y. Wang, J. Ren, L. Pan, W. Wu, L. Yang, J. Wang, C. Qian, *et al.*, "Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 803–814, 2023.

[43] Sketchfab, "Sketchfab: 3d models platform." `https://sketchfab.com`. Accessed: 2025-09-10.

[44] TurboSquid, "Turbosquid: 3d models marketplace." `https://www.turbosquid.com`. Accessed: 2025-09-10.

[45] CGTrader, "Cgtrader: 3d models for vr / ar." `https://www.cgtrader.com`. Accessed: 2025-09-10.

[46] Free3D, "Free3d: Free 3d models." `https://free3d.com`. Accessed: 2025-09-10.

[47] GrabCAD, "Grabcad community library." `https://grabcad.com`. Accessed: 2025-09-10.

[48] J. Hang, X. Lin, T. Zhu, X. Li, R. Wu, X. Ma, and Y. Sun, "Dexfuncgrasp: A robotic dexterous functional grasp dataset constructed from a cost-effective real-simulation annotation system," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, pp. 10306–10313, 2024.

[49] G. Pavlakos, D. Shan, I. Radosavovic, A. Kanazawa, D. Fouhey, and J. Malik, "Reconstructing hands in 3d with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9826–9836, 2024.

A More Experiment Results

In this section, we present additional qualitative visualizations in Figure 7, where representative frames are sampled from generated manipulation sequences. These examples cover four settings: i) seen objects during training, ii) unseen objects within a category, iii) unseen functional parts within a category, and iv) unseen categories. Functional parts are highlighted in red for clarity. The results show that our method can synthesize diverse manipulation for the same functional part, as illustrated by different ways of using a camera. For objects with multiple functional parts, such as a mouse, our method further produces part-aware interactions tailored to distinct functional regions. These examples demonstrate the flexibility of our approach and its ability to generalize across both unseen functional parts and unseen object categories.

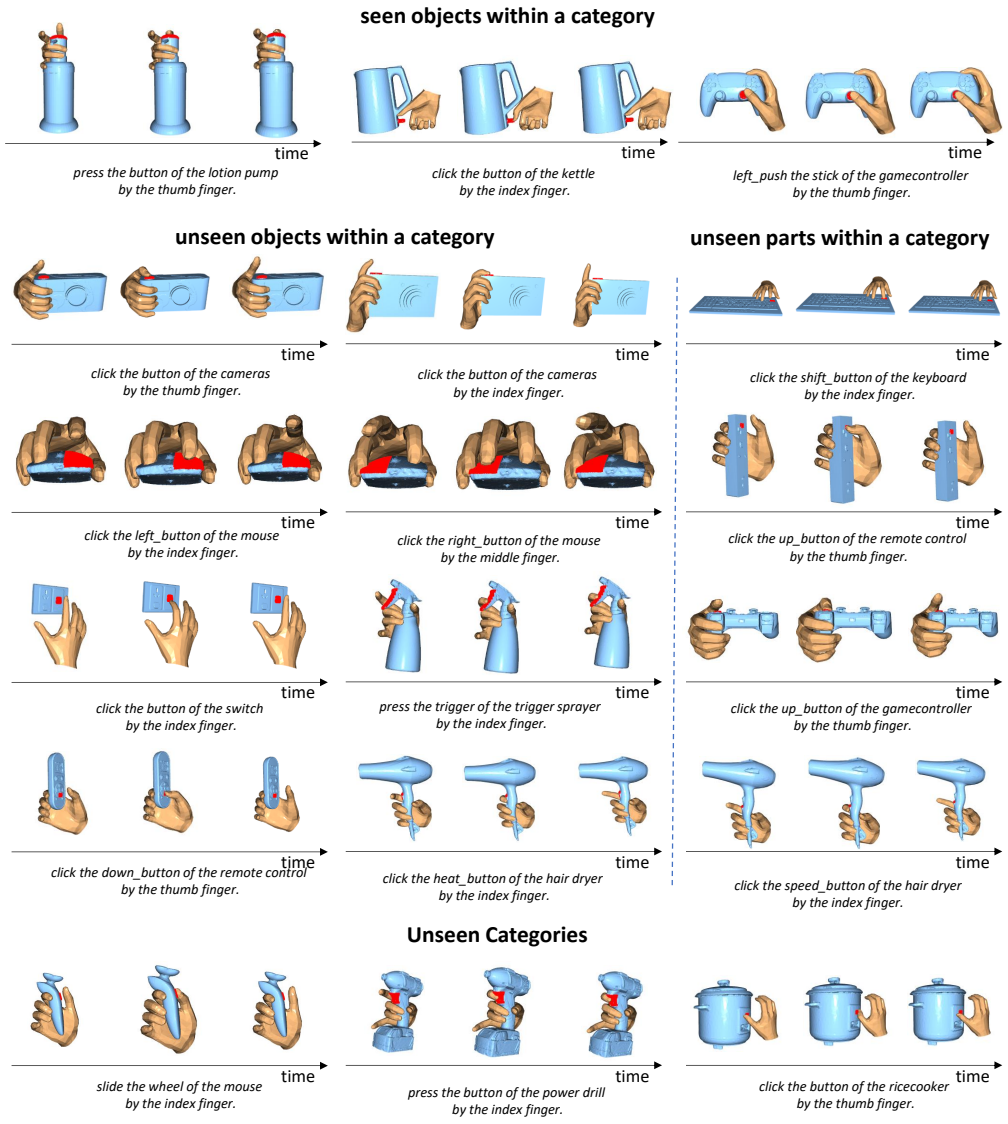


Figure 7: More qualitative visualizations.

B More Dataset Details

In this section, we provide additional details about our dataset. We construct a fine-grained functional manipulation dataset covering multiple object categories. Specifically, we collect objects from 20

Table 4: Object categories grouped by functional-part complexity

Type	Categories	#Categories	#Instances
Single functional part	cameras, display, trigger_sprayer, electric_clippers, electric_shaver, laptop, lotion_pump, umbrella, power_drill, rice_cooker, electric_toothbrush, stapler, switch, kettle, flashlight	15	483
Multi functional part	remote_control, gamecontroller, hair_dryer, keyboard, mouse	5	199

Table 5: Categories with multiple functional parts

Object Category	Functional Parts	Manipulation Finger
remote_control	power_button, up_button, down_button, left_button, right_button	thumb
gamecontroller	stick, up_button, down_button, left_button, right_button, trigger	thumb, index
hair_dryer	heat_button, speed_button	index
keyboard	space_button, shift_button, capslock_button, enter_button, backspace_button	thumb, index, middle
mouse	left_button, right_button, wheel	index, middle

categories using public resources. As shown in Table 4, the dataset includes 15 categories with a single functional part, such as cameras and trigger sprayer, comprising 483 instances in total, and 5 categories with multiple functional parts, such as remote controllers and game controllers, comprising 199 instances. The functional parts of these objects are listed in Table 5. For example, a game controller contains multiple buttons that can be operated by different fingers. Such diversity in object categories and functional parts ensures the richness and variety of our dataset. We then follow the core idea of DexFuncGrasp [48] and extend it to sequential manipulation data acquisition. As illustrated in Figure 8, each object is instantiated in a virtual environment. We use a monocular RGB camera (RealSense) to capture images in real time and employ the high-accuracy hand pose estimator HaMeR [49] to reconstruct hand pose sequences, which are then synchronized and visualized in the virtual environment for interaction with the objects. Since HaMeR estimates hand pose from monocular RGB images, the wrist joint is often inaccurate, leading to an overly large hand motion space and making data collection difficult. To address this issue, we instead control the wrist position using a keyboard. During the process, the data must be checked for both physical plausibility and task completion. Collection can continue only after both checks are passed. This real-to-virtual pipeline enables cost-effective collection of functional manipulation data. In addition, for each annotated task, we require annotators to provide at least three different manipulation gestures for the object, so as to ensure the diversity of the dataset. In total, we collect 1,879 manipulation sequences, each annotated with object point clouds, functional parts, task descriptions, object category labels, and corresponding MANO-based [34] hand motion sequences.

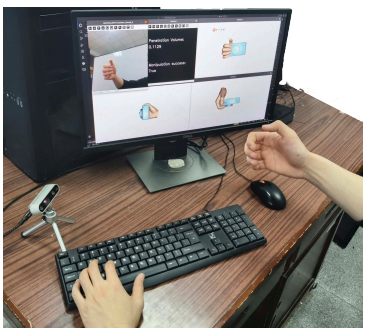


Figure 8: Data collection platform.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims of the paper are clearly stated in the Abstract and Introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations of the proposed method are discussed in Section 5 (Limitations and Future Work).

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: The paper does not include theoretical results requiring formal assumptions or proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides sufficient details to reproduce the experimental results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The code repository is not publicly released at submission time and the authors intend to release code upon acceptance.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies all the training and test details in Section 3.4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We did not report error bars for experiments because repeating the full training runs under multiple random seeds would require substantially more compute. Instead, we report results under a fixed experimental protocol and include detailed hyperparameters, data splits, and compute information to support reproducibility.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper provides the main compute resource information used in the experiments in Section 3.4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The authors have reviewed the NeurIPS Code of Ethics and confirm that the research complies with its requirements.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses the societal impact of the proposed method. Positively, the work can facilitate research in precise manipulation synthesis. We do not anticipate significant negative societal impacts, as the dataset and method are intended for academic research and evaluation.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This work does not release high-risk models, scraped datasets, or other assets that would require special safeguards for controlled release.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper properly credits all existing datasets, models, and software assets used in the experiments. We follow the original licenses and terms of use to the best of our knowledge, and relevant citations and URLs are included in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces a new dataset and provide detailed descriptions in Section 4.1 and Appendix.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments and research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing research.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The paper employs large language models for the understanding of task texts, and the specific usage of LLMs has been fully and clearly described in the paper.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.