Dexonomy: Synthesizing All Dexterous Grasp Types in a Grasp Taxonomy

Jiayi Chen[1,2*], Yubin Ke[1,2*], Lin Peng[2] and He Wang[1,2,3†]

Fig. 1: For **any grasp type** in GRASP taxonomy [1], **any object**, and **any articulated hand**, our pipeline efficiently synthesizes contact-rich, penetration-free, and physically plausible dexterous grasps, starting from only one human-annotated grasp template to specify an initial hand pose and contact information per hand and grasp type.

***Abstract*— Generalizable dexterous grasping with suitable grasp types is a fundamental skill for intelligent robots. Developing such skills requires a large-scale and high-quality dataset that covers numerous grasp types (i.e., at least those categorized by the GRASP taxonomy), but collecting such data is extremely challenging. Existing automatic grasp synthesis methods are often limited to specific grasp types or object categories, hindering scalability. This work proposes an efficient pipeline capable of synthesizing contact-rich, penetration-free, and physically plausible grasps for any grasp type, object, and articulated hand. Starting from a single human-annotated template for each hand and grasp type, our pipeline tackles the complicated synthesis problem with two stages: optimize the object to fit the hand template first, and then locally refine the hand to fit the object in simulation. To validate the synthesized grasps, we introduce a contact-aware control strategy that allows the hand to apply the appropriate force at each contact point to the object. Those validated grasps can also be used as new grasp templates to facilitate future synthesis. Experiments show that our method significantly outperforms previous type-unaware grasp synthesis baselines in simulation. Using our algorithm, we construct a dataset containing 10.7k objects and 9.5M grasps, covering 31 grasp types in the GRASP taxonomy. Finally, we train a type-conditional generative model that successfully performs the desired grasp type from single-view object point clouds, achieving an 82.3% success rate in real-world experiments. Project page: https://pku-epic.github.io/Dexonomy.**

[1]Peking University. [2]Galbot. [3]Beijing Academy of Artificial Intelligence.
*Equal contribution. †Corresponding author: hewang@pku.edu.cn.

I. Introduction

Dexterous grasping is a fundamental skill for intelligent robots, enabling flexible interaction with the environment. However, most prior work focuses on whether a dexterous hand can successfully grasp an object, rather than considering *how* to grasp it. As a result, the dexterous hand loses its dexterity and becomes functionally similar to a large parallel gripper. True dexterous grasping is not merely about “grasping with dexterous hands", but about “grasping dexterously with appropriate grasp types based on the task requirement". For example, when a robot needs to securely grasp an apple or hold a knife to cut, it should use a power grasp to envelop the object. Conversely, when grasping a lightweight or flat object from the table, a precision grasp using the fingertips would be more suitable.

To develop such intelligent skills, there are two key challenges: (1) selecting the appropriate grasp type based on the task and (2) generating high-quality grasps for specified types and objects. The first challenge is a high-level decision-making problem and can take advantage of recent advances in large vision-language models, e.g., GPT-4o [2], as a temporary solution. However, the second challenge is less studied and represents a significant bottleneck, which is the main focus of this paper. To address the problem of type-

aware grasp synthesis with data-driven methods, the first step is to build a large-scale grasp dataset that at least includes most of the grasp types in the GRASP taxonomy [1]. However, collecting grasp data, particularly for multi-fingered hands in contact-rich scenarios, remains a big challenge.

Several approaches have been explored for automatically synthesizing a large-scale dexterous grasp dataset, but most of them suffer from various limitations. Analytical grasp synthesis methods [3], [4], [5], [6], [7] are often applicable to any object, but most of them are type-unaware and the synthesized grasps only belong to limited types. This is because specifying flexible grasp types solely through analytical metrics is challenging. Moreover, these methods often produce unnatural hand poses, as they prioritize *force closure*, which does not always align with human habits. Another line of research [8], [9], [10] focuses on transferring functional dexterous grasps by mapping object contact regions. While these methods generate more human-like grasps and support a wider range of grasp types, they are limited to objects that are geometrically similar or axis-aligned with the initial demonstration, making them less scalable.

In this work, we propose a novel pipeline to address these challenges. As shown in Figure 1, our algorithm can efficiently synthesize high-quality dexterous grasps for any grasp type, object, and articulated hand, requiring only one human-annotated grasp template per hand and grasp type. Our synthesized grasps achieve rich hand-object contact (e.g., > 10 hand links within 2 mm of the object for power grasps), guarantee penetration-free poses via collision mesh verification, and satisfy force closure under six-axis testing in MuJoCo [11] — all with shared hyperparameters across grasp types, objects, and hands.

Our key insight is that grasping can be framed as a geometric matching problem, where the hand and object should align through contact points. We begin by introducing a human-annotated grasp template that specifies the initial hand pose and contact information (i.e., points and normals). Unlike previous methods that directly adjust the hand pose to fit the object, we first sample and optimize the object pose to match the hand contacts defined in the grasp template. This stage supports hundreds of thousands of initial samples processed in parallel on a single GPU and leaves only a small number of promising results for the next stage.

After aligning the object pose, the hand only needs a slight refinement to get a good grasp. This dual-stage design not only eases the hand refinement, but also ensures that the final grasp remains similar to the initial grasp template and thus remains natural. In contrast to most prior work [12], [3], [6], [7] that develops custom objective functions and optimizers to refine the hand pose, we propose a novel method based on the transposed Jacobian control in MuJoCo. This approach is key to achieving rich contacts while ensuring no penetration, with minimal coding effort and parameter tuning.

Next, we evaluate the synthesized grasps in MuJoCo to assess their ability to withstand external forces applied to the object. Unlike previous work [3], [13] that designs heuristics to squeeze the hand for applying force on the object, which is not suitable for all grasp types, we design a contact-aware control strategy that computes the desired forces for each contact point and controls the hand to apply them approximately, also based on the transposed Jacobian control. Finally, high-quality grasps that pass the simulation tests can be used to construct new grasp templates, reducing the need for human annotations and broadening the range of objects that can be grasped.

Experiments show that our method greatly outperforms previous type-unaware grasp synthesis baselines in simulation. Using our proposed pipeline, we also build a large-scale dataset covering different grasp types. This dataset further enables training a type-conditional generative model that generates desired grasp types for novel objects from single-view point clouds, achieving a success rate of 82.3% on the Shadow hand in real-world experiments. Finally, we show that our algorithm can be used to develop an annotation UI for collecting semantic grasps on the specified object regions with only a few mouse clicks.

In summary, our main contributions are:

- An efficient pipeline to synthesize high-quality grasps for any grasp type, object, and hand, starting from one human-annotated template per hand and grasp type.
- A large-scale dataset with 9.5M grasps and 10.7k objects, covering 31 grasp types in the GRASP taxonomy.
- A type-conditional generative model that can use the specified grasp types to grasp novel objects in the real world, with only a single-view point cloud as input.
- An annotation UI for collecting semantic grasps with only a few mouse clicks.

II. Method

A. Grasp Template Definition

A grasp template consists of several components: the hand joint configuration $\mathbf{q} \in \mathbb{R}^q$, hand contact points $\mathbf{p}_i^h \in \mathbb{R}^3$, corresponding normals $\mathbf{n}_i^h \in \mathbb{R}^3$, and the link name for each contact point ($i = 1, 2, \ldots, m$). Our algorithm requires a single human-annotated grasp template for each hand and grasp type as initialization.

B. Lightweight Global Alignment of Object Pose

In this stage, we simultaneously sample and optimize the object pose to align with the selected template's hand contacts while keeping the hand pose fixed. The optimization variable is the object's transformation, parameterized by its scale $s_o \in \mathbb{R}$, rotation $\mathbf{R}_o \in \mathcal{S}^3$, and translation $\mathbf{t}_o \in \mathbb{R}^3$.

Before optimization, we begin with dense sampling. First, a random grasp template is selected from the *Grasp Template Library*, and a random hand contact from the template is chosen. Then, a random object is selected, and a random surface point on the object is chosen. The object is initialized by aligning the sampled hand and object contacts, where contact points are matched and the contact normal directions are set opposite. The object's scale and in-plane rotation perpendicular to the normal direction are sampled randomly. Our pipeline supports parallelizing massive samples of different contacts, objects, and grasp templates on a single GPU.

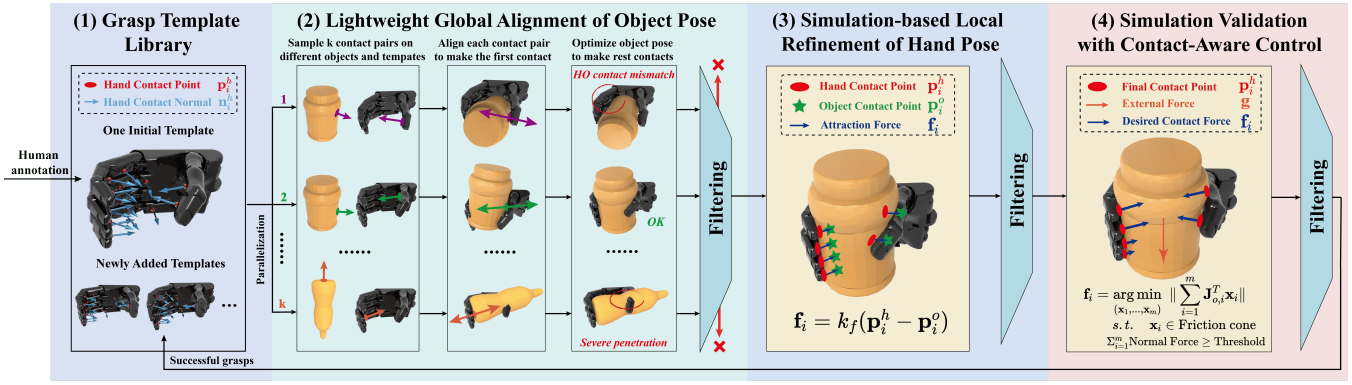

Fig. 2: **The pipeline of Dexonomy.** (1) *Grasp Template Library* initially requires one human-annotated template. (2) *Lightweight Global Alignment* stage samples and optimizes the object poses in parallel on a GPU, to match the contact points and normals of the selected grasp templates. (3) *Simulation-based Local Refinement* stage adjusts the hand pose to improve hand-object contacts. (4) *Simulation Validation* tests force-closure grasps using our proposed contact-aware control strategy. (5) New templates are constructed from successful grasps and added to the *Grasp Template Library*, used in the following iterations.

During each optimization iteration, each hand contact point $\mathbf{p}_i^h$ calculates the nearest point $\mathbf{p}_i^o$ on the object's surface using the differentiable library `Warp` [14]. To penalize the mismatch between hand and object contacts, we optimize the object pose by minimizing the following energy function:

$$L = k_p \sum_{i=1}^{m} \|\mathbf{p}_i^h - \mathbf{p}_i^o\|^2 + k_n \sum_{i=1}^{m} \|\mathbf{n}_i^h - \mathbf{n}_i^o\|^2 \qquad (1)$$

where k_p and k_n are hyperparameters. There is no other energy used for optimization except Eq. 1.

After optimization, results are filtered using four criteria. First, the final energy function must be below a threshold to ensure a good match between hand-object contacts. Second, severe penetration between the hand and object should be avoided, which we efficiently detect using our proposed hand collision skeletons parameterized by line segments (details in SUPP). Third, the object contact quality, as measured by the QP-based grasp energy from BODex [7], must exceed a threshold. Finally, we apply a process similar to farthest point sampling to filter out duplicate object transformations.

Our design, using only one energy during optimization and leaving other checks for post-filtering, provides several advantages. First, it reduces computational costs, enabling maximized parallelization to benefit from dense sampling to avoid local optimum traps. Second, it reduces sensitivity to hyperparameters, as filtering criteria are applied sequentially, while optimization energies need to be applied together.

C. Simulation-based Local Refinement of Hand Pose

In this stage, the object is fixed, and the hand pose is locally refined to improve the hand-object contact. A virtual force $\mathbf{f}_i$ is needed at each hand point $\mathbf{p}_i^h$ toward the corresponding nearest object point $\mathbf{p}_i^o$. To apply these virtual forces in MuJoCo, they are transferred to the hand's joint torque via simplified transposed Jacobian control:

$$\mathbf{f}_i = k_f(\mathbf{p}_i^h - \mathbf{p}_i^o), \quad \tau = \sum_{i=1}^{m} \mathbf{J}_{h,i}^T \mathbf{f}_i \qquad (2)$$

where k_f is a hyperparameter and $\mathbf{J}_{h,i}^T \in \mathbb{R}^{q\times 3}$ is the transpose of the hand contact Jacobian that maps force vectors from the world to joint coordinates.

While Eq. 2 is a simplified control strategy with many assumptions (e.g., no dynamics or gravity; joint torques mapped from each contact force are independent and additive), it serves our need for synthesizing contact-rich grasps in simulation. This is easy to implement and works for other physics simulators. Eq. 2 is iteratively applied for 200 steps, with $\mathbf{p}_i^h$ remaining static in the hand link frame and $\mathbf{p}_i^o$ remaining static in the world frame to avoid drift.

After optimization, we filter the results based on three criteria. First, there should be no hand-object penetration, measured using collision meshes. Second, all fingers that have at least one annotated contact should touch the object, meaning the minimal distance between hand links and object meshes should be within 2 mm. Finally, the grasp quality must exceed a threshold, as in the previous stage.

D. Simulation Validation with Contact-Aware Control

To validate the synthesized grasps in MuJoCo, the hand should squeeze to hold the object stably, controlled by a control signal of joint torques. Our contact-aware control strategy first calculates the desired forces on each contact using the quadratic programming (QP) [7], and then converts these forces into joint torques using the same transposed Jacobian control as in Eq. 2. A grasp is considered to succeed only if the object remains stable under all six orthogonal external forces for 2 seconds in simulation.

E. Construction of New Grasp Templates

Once a grasp successfully passes the simulation validation, a new grasp template is constructed and added to the template library. The joint configuration of the new template is taken from the successful grasp, while the contact information is updated only if an actual contact is detected near the original contact on the same hand link. This strategy prevents the new template's contact information from deviating too much

	GSR (%) ↑	OSR (%) ↑	S (s^{-1}) ↑	CLN ↑	CDC (mm) ↓	PD (mm) ↓	SPD (mm) ↓	D (%) ↓
DexGraspNet [3]	12.10	57.01	3.25	3.22	7.58	4.85	1.20	29.03
FRoGGeR [5]	10.34	55.70	2.98	2.51	4.95	0.22	**0.00**	**27.01**
SpringGrasp [6]	7.83	35.44	5.47	2.79	23.59	16.58	1.06	70.18
BODex [7]	49.23	**96.56**	**403.9**	3.85	3.03	0.63	0.02	32.50
Ours	**60.50**	96.53	323.4	**4.38**	**0.21**	**0.00**	**0.00**	34.17

TABLE I: **Comparison with Type-Unaware Grasp Synthesis Baselines for Allegro Hand in Simulation.** Most baselines, except DexGraspNet, only synthesize fingertip grasps, so we also synthesize fingertip grasps for a fair comparison.

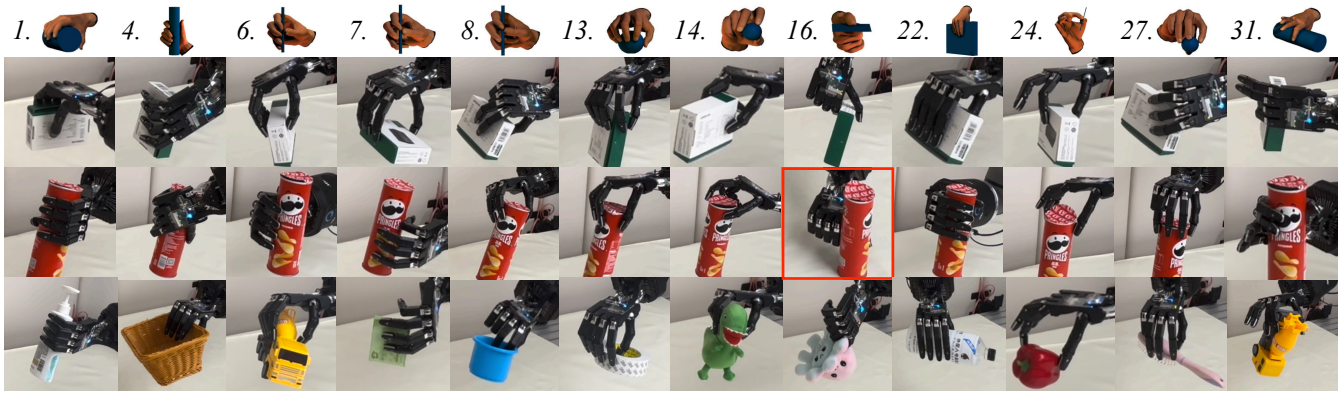

Fig. 3: **Real-World Gallery.** Our trained type-conditional generative model synthesizes desired grasp types from single-view object point clouds. All grasps succeed except the one in the red box, where the grasp type is unsuitable for the object.

from the original. Newly added templates can be randomly selected in the global alignment stage of the following loops.

III. Experiment

A. Type-Unaware Grasp Synthesis in Simulation

Although our work focuses on type-aware dexterous grasp synthesis, there is no suitable baseline available for direct comparison. Therefore, we conduct experiments on type-unaware grasp synthesis in simulation to demonstrate the effectiveness of our pipeline.

Evaluation metrics. Eight metrics similar to BODex [7] are used for a comprehensive evaluation: Grasp Success Rate (GSR), Object Success Rate (OSR), Speed (S), Contact Link Number (CLN), Contact Distance Consistency (CDC), Penetration Depth (PD), Self-Penetration Depth (SPD), Diversity (D). The detailed description of each metric is in SUPP.

Experiment setup. We use the Allegro hand and 5689 object assets from DexGraspNet, with six scales applied to each normalized object: 0.05, 0.08, 0.11, 0.14, 0.17, and 0.20. Each method allows 20 attempts, where for our method one attempt is defined as one valid result output by the global alignment stage. Our reported speed does not include simulation validation for a fair comparison with baselines, and the detailed time analysis is in SUPP.

Result analysis. As shown in Table I, our method achieves the highest grasp success rate and best performance on contact and penetration. The penetration for our grasps is consistently 0 because we set a 1mm contact margin in MuJoCo, and MuJoCo can resolve millimeter-level penetration. Our speed is slightly lower than that of BODex, as their pipeline mainly runs on GPUs, while our local refinement stage uses MuJoCo's CPU version. Our diversity is somewhat lower, as we use only two similar templates and a smaller step number for refining hand poses compared to the baselines. However, the overall diversity of our synthesized grasps for all grasp types is much better, as reported in SUPP. The success rates of baselines are lower than those reported in BODex [7], primarily because our objects have a higher mass (100g vs. 30g) and a larger scale range ($[0.05, 0.2]$ vs. $[0.06, 0.12]$).

B. Learning Type-Aware Grasp Synthesis

Using our proposed method, we generate a large-scale dataset for Shadow Hand covering 31 grasp types in the GRASP taxonomy. We also propose a type-conditional generative model based on normalizing flow [15], [7]. The main idea is just to add a conditional codebook, where each grasp type corresponds to a code in it. Since the learning model is just used as proof-of-concept and not the main contribution of this paper, the details are left in SUPP.

To perform a grasp, a single-view object point cloud segmented by SAM2 [16] and the specified grasp type are taken as input to the trained type-conditional generative model. The model generates 100 candidates and we use the pre-grasp poses as the target for collision-free motion planning with CuRobo [17], filtering out failed ones. The remaining grasps are ordered by the output probability of the normalizing flow, and the top 3 are executed. In this way, we prevent the success rate of motion planning from affecting the results, since it is not the focus of this paper. After reaching the pre-grasp pose, the hand moves to the grasp pose and then the squeeze pose to grasp the object stably, and finally lift it. As shown in Fig. 3, our model can correctly generate physically plausible grasps for the specified types and achieves an overall success rate of 82.3% on 13 test objects.

REFERENCES

[1] 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.

[2] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, *et al.*, "Gpt-4o system card," *arXiv preprint arXiv:2410.21276*, 2024.

[3] R. Wang, J. Zhang, J. Chen, Y. Xu, P. Li, T. Liu, and H. Wang, "Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 11 359–11 366.

[4] D. Turpin, T. Zhong, S. Zhang, G. Zhu, E. Heiden, M. Macklin, S. Tsogkas, S. Dickinson, and A. Garg, "Fast-grasp'd: Dexterous multi-finger grasp generation through differentiable simulation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 8082–8089.

[5] A. H. Li, P. Culbertson, J. W. Burdick, and A. D. Ames, "Frogger: Fast robust grasp generation via the min-weight metric," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 6809–6816.

[6] S. Chen, J. Bohg, and C. K. Liu, "Springgrasp: An optimization pipeline for robust and compliant dexterous pre-grasp synthesis," *arXiv preprint arXiv:2404.13532*, 2024.

[7] J. Chen, Y. Ke, and H. Wang, "Bodex: Scalable and efficient robotic dexterous grasp synthesis using bilevel optimization," *arXiv preprint arXiv:2412.16490*, 2024.

[8] 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*, 2022, pp. 20 953–20 962.

[9] W. Wei, P. Wang, S. Wang, Y. Luo, W. Li, D. Li, Y. Huang, and H. Duan, "Learning human-like functional grasping for multi-finger hands from few demonstrations," *IEEE Transactions on Robotics*, 2024.

[10] R. Wu, T. Zhu, X. Lin, and Y. Sun, "Cross-category functional grasp tansfer," *arXiv preprint arXiv:2405.08310*, 2024.

[11] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2012, pp. 5026–5033.

[12] H. Jiang, S. Liu, J. Wang, and X. Wang, "Hand-object contact consistency reasoning for human grasps generation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 11 107–11 116.

[13] J. Zhang, H. Liu, D. Li, X. Yu, H. Geng, Y. Ding, J. Chen, and H. Wang, "Dexgraspnet 2.0: Learning generative dexterous grasping in large-scale synthetic cluttered scenes," in *8th Annual Conference on Robot Learning*, 2024.

[14] M. Macklin, "Warp: A high-performance python framework for gpu simulation and graphics," https://github.com/nvidia/warp, March 2022, nVIDIA GPU Technology Conference (GTC).

[15] Y. Xu, W. Wan, J. Zhang, H. Liu, Z. Shan, H. Shen, R. Wang, H. Geng, Y. Weng, J. Chen, *et al.*, "Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4737–4746.

[16] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, *et al.*, "Sam 2: Segment anything in images and videos," *arXiv preprint arXiv:2408.00714*, 2024.

[17] B. Sundaralingam, S. K. S. Hari, A. Fishman, C. Garrett, K. Van Wyk, V. Blukis, A. Millane, H. Oleynikova, A. Handa, F. Ramos, *et al.*, "Curobo: Parallelized collision-free robot motion generation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 8112–8119.