RoboSynChallenge: Mastering Real-World Dexterity via Generalizing Synthesized Manipulation Skills

Runyi Zhao[1,2], **Ruixin Wu**[1,2]†, **Hongrui Zhang**[1,2]†, **Chengkun Li**[1,2]†,
Ang Li[2], Ruixing Jin[2], Yueci Deng[2,11], Yingying Guo[2],
Lihe Ding[3], Shaocong Dong[4], Tianfan Xue[3], Yanjun Gao[5], Yudong Luo[6],
Pascal Poupart[7,8], Simo Wu[9], Kui Jia[2,11], Wei-shi Zheng[1,10], Guiliang Liu[1,2]

[1]Shenzhen Loop Area Institute (SLAI) [2] The Chinese University of Hong Kong, Shenzhen
[3] The Chinese University of Hong Kong [4] The Hong Kong University of Science and Technology
[5] LARK Lab, University of Colorado Anschutz [6] Mila - Quebec AI Institute, Canada [7] Vector Institute
[8] University of Waterloo [9] Fudan University [10] Sun Yat-sen University [11] DexForce

robosynchallenge@gmail.com

● **Website**: https://robosyn-bench.net/
◆ **GitHub**: https://github.com/EDEM-AI/RoboSynChallenge/
✍ **Tutorial**: https://edem-ai.github.io/RoboSynChallenge/html/

Abstract

Achieving generalizable robotic manipulation remains a central challenge in embodied intelligence. Despite rapid advances in model architectures and learning algorithms, progress is often limited by the scarcity and narrow diversity of real-world data. The **RoboSynChallenge** competition introduces a unified benchmark to evaluate and advance the *generalizability* of manipulation policies across a spectrum of tasks, environments, and difficulty levels. To alleviate the shortage of realistic data, the challenge integrates large-scale synthetic data generation with standardized real-world robotic evaluation. Participants are encouraged to leverage synthesized state-action trials to improve general-purpose policy learning, while final assessments are conducted exclusively on unseen real-world manipulation environments. Baseline implementations, including Transformer-, Diffusion-, Vision-Language-Action, and World-Action-Model–based policies, are provided to ensure reproducibility and comparability. By coupling scalable simulation-based training with rigorous real-world validation, **RoboSynChallenge** aims to foster the development of broadly capable, data-efficient, and adaptable manipulation systems, thereby paving the way toward truly general robotic intelligence.

1 Competition description

Background. Developing scalable and precise robotic manipulation policies represents a critical step toward realizing the vision of *Embodied Artificial Intelligence (EAI)* [1]. As a technology with broad societal and industrial impact, scalable robotic manipulation holds the potential to reduce operational costs and enhance productivity across manufacturing, household, and healthcare domains [2].

Data-driven learning methods have emerged as a principled approach for developing general-purpose robotic agents that can operate flexibly across diverse tasks and environments. In recent years, *generalist policy models* have been advanced through the use of Transformers [3], diffusion models [4], Vision-Language-Action (VLA) frameworks [5], and World Action Models (WAMs) [6, 7, 8, 9]. These architectures enable end-to-end learning pipelines that map multimodal observations and language instructions directly to continuous robot actions [10], paving the way for more autonomous, instruction-following control systems.

Preprint.

As research progresses toward developing generalist manipulation policies, a central question arises: how can we fairly and comprehensively evaluate and benchmark such general-purpose robotic systems? Existing benchmarks are typically categorized as either simulation-based [11, 12, 13, 14] or real-world [15, 16, 17, 18]. However, unlike vision or language models, which can draw upon vast internet-scale datasets, real-world robotic benchmarks rely on data obtained through physical interaction, making data collection costly, slow, and tied to specific hardware setups [5]. While simulation facilitates scalable data generation [19, 20, 21, 22], mismatches in dynamics, kinematics, and sensing inevitably yield a simulation-to-reality (Sim2Real) gap, resulting in degraded real-world performance [14, 23, 24]. Bridging this Sim2Real gap thus remains a fundamental challenge in embodied intelligence research. Moreover, most existing benchmarks rely on fixed datasets, which assume static and controlled environments. This design misaligns with the objective of building generalist policies that must generalize to unseen objects and diverse environments.

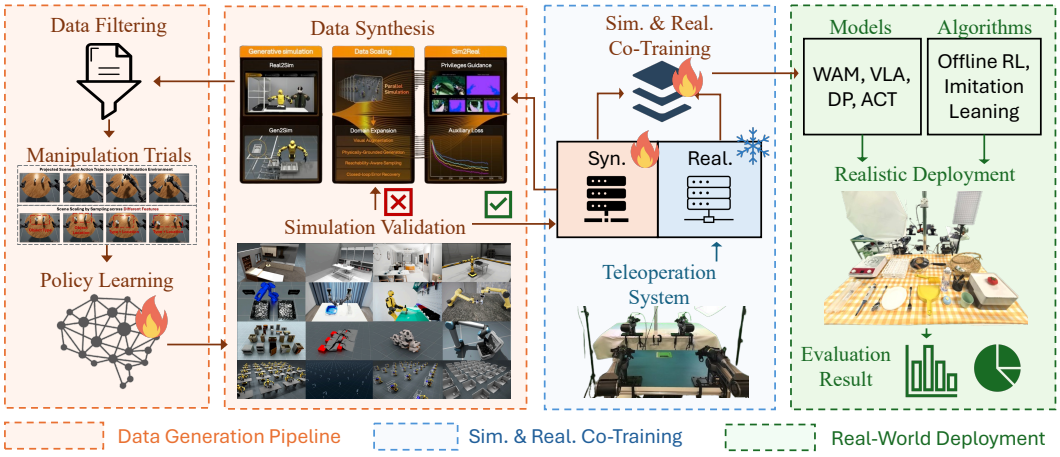

Figure 1: The pipeline of RoboSynChallenge, which provides a comprehensive data generation framework for synthesizing manipulation trials within a simulated environment (in orange background), thereby expanding the training datasets. These synthetic data, together with a smaller amount of static real-world data collected through teleoperation (in blue background), can be used for simulation-to-real co-training. This approach helps mitigate the limitations of scarce real data when deploying trained manipulation policies in the real world (in green background).

Impact. In this proposal, we push the frontier a step further by introducing RoboSynChallenge, a competition designed to quantify *how effectively simulated data can improve the generalizability of manipulation policies when real-world data is scarce*. To achieve this goal, RoboSynChallenge provides a comprehensive data-generation and evaluation pipeline that integrates scalable simulation environments with real-world physical setups for performance validation. The simulation platform enables practitioners to flexibly configure and scale dataset generation across diverse conditions, while the real-world evaluation protocol encompasses multiple levels of difficulty. Together, these components facilitate a rigorous and systematic analysis of Sim2Real transferability.

The RoboSynChallenge is envisioned as an open and scalable benchmark to advance research on generalizable robot learning across simulated and real-world domains. The anticipated impact of RoboSynChallenge is multifaceted as follows:

1) Standardized Benchmark. RoboSynChallenge seeks to establish a standardized real-world benchmark that spans the entire dexterous manipulation policy pipeline, from large-scale data generation and policy learning with synthesized data, to rigorous evaluation under physical environments.

2) Real-world Reproducibility. RoboSynChallenge strengthens the reproducibility of robotic control policies through standardized data synthesis protocols and evaluation frameworks, ensuring consistent benchmarking and fair comparison across real-world environments.

3) Opensource Tools and Baselines. RoboSynChallenge provides publicly available baseline models, data synthesis, and training pipelines, fostering transparent evaluation, reproducible experimentation, and collaborative progress within the robotics research community.

4) Toward Generalizable Manipulation. RoboSynChallenge provides a foundational platform toward generalizable manipulation policies by coupling large-scale synthetic data generation with standardized real-world evaluation, enabling systematic study of scalable policy generalization.

1.1 Novelty

As a new competition, RoboSynChallenge introduces substantial differences compared to previous and ongoing challenges. Our key innovations are as follows:

1) Benchmarking Sim2Real Transferability. RoboSynChallenge aims to fill a critical gap in the field by providing the first standardized benchmark for Sim2Real transferability. It systematically evaluates how well simulated data and policies support scaling to complex, real-world environments. Beyond measuring raw performance, RoboSynChallenge assesses robustness, adaptability, and generalization under real-world noise and domain shifts. This approach fundamentally differs from prior Sim2Sim or Real2Real benchmarks that operate within a single modality. It represents a significant step toward unified Sim2Real evaluation and understanding the limits of simulation-based learning.

2) Generative Data Streaming. Unlike prior benchmarks that rely on a fixed, human-crafted dataset, RoboSynChallenge integrates an automated data generation pipeline [25]. The system procedurally synthesizes simulation environments and automatically generates motion trails in a closed loop. Moreover, to bridge the Sim2Real gap, practitioners can use the provided realistic data as input to style transfer models [26], video generation models [27], or scene augmentation techniques [28, 19] to synthesize diverse motion trajectories with high visual fidelity. This enables large-scale, diverse, and efficient data acquisition without manual curation.

3) Real-World Evaluation. RoboSynChallenge integrates real-world robotic evaluation as part of its official leaderboard. Submitted policies are deployed and tested on standardized physical robot platforms using the same task definitions and evaluation metrics as in simulation. This unified testbed directly measures Sim2Real performance and provides a transparent benchmark for real-world robustness, enabling participants to assess both efficiency and adaptability under tangible constraints.

Table 1 highlights the distinctions between our approach and existing benchmarks. Unlike previous benchmarks that were limited to single-arm settings, simulations, or real-world datasets alone, RoboSynChallenge unifies bimanual manipulation across both simulated and realistic environments. Crucially, it leverages simulated data in a generative manner to augment real-world interactions, enabling richer policy learning and improved generalization. This hybrid paradigm bridges the data efficiency of simulation with the robustness of real-world performance, extending the scope of dexterous manipulation beyond existing competitive and benchmark platforms.

Table 1: The comparison to other competition and benchmarks for dexterous manipulation.

Competition	Rigid Objects	Articulated Objects	Assembling Objects	Tool Using	Manip. Setting	Dataset	Training Env.	Evaluation Env.
RLBench [11]	✓	✓	✗	✗	Single-Arm	Offline	Simulated	Simulated
CALVIN [12]	✓	✓	✗	✗	Single-Arm	Offline	Simulated	Simulated
LIBERO [13]	✓	✓	✗	✗	Single-Arm	Offline	Simulated	Simulated
RoboTwin [29]	✓	✓	✗	✓	Both	Offline	Simulated	Simulated
WBCD [1]	✓	✓	✓	✗	Bimanual	None	Realistic	Realistic
RoboChallenge [15]	✓	✓	✓	✓	Both	Offline	Realistic	Realistic
RobotArena ∞ [16]	✓	✓	✗	✗	Single-Ar	None	Realistic	Realistic
RoboArena [17]	✓	✓	✗	✓	Single-Arm	Offline	Realistic	Simulated
ManipulationNet [18]	✓	✗	✓	✗	Single-Arm	Offline	Realistic	Realistic
ManipArena [2]	✓	✓	✗	✗	Bimanual	Offline	Realistic	Realistic
RoboSynChallenge	✓	✓	✓	✓	Bimanual	Generative	Sim. & Real.	Realistic

1.2 Data

The dataset employed in RoboSynChallenge is primarily generated specifically for this competition using the *EmbodiChain* [25] generative simulation framework. It comprises large-scale embodied interaction trajectories, multi-modal sensory streams (including RGB-D data, proprioceptive signals, and physics-based scene metadata), and structured annotations detailing task semantics, contact events, and success outcomes. To moderate the difficulty of Sim2Real transfer, RoboSynChallenge also includes a smaller subset of data collected through teleoperation to provide real-world correspondence for a limited set of scenarios. The detailed generation process is described as follows.

All robot data were produced within robot manipulation environments using a simulation environment, generative methods, and teleoperation. No personally identifiable or human data are included.

Data Collection Protocol. To facilitate the generalizability of manipulation policy in the RoboSyn-Challenge, we designed protocols for collecting both simulated and real-world datasets.

1) *Real-World Data Collection Protocol.* To construct the real-world dataset, each manipulation task was conducted under five distinct experimental conditions, each further evaluated across four positional variations and three orientation settings, resulting in a total of 60 samples per task. The experimental conditions are designed to introduce variability in background texture, illumination, and scene complexity, thereby enabling a rigorous assessment of generalization in real-world manipulation scenarios. The detailed configuration of the experimental conditions is summarized in Table 2 in Appendix A.1. Each condition was combined with four positional variations and three orientation settings, ensuring comprehensive coverage of spatial and visual factors across all collected samples.

2) *Synthesized Data Collection Protocol.* RoboSynChallengeintegrates Embodichain [25] to scale up the simulated state-action trails. Appendix A.2 shows the details of data generation. Each simulation instance involved systematic perturbations in lighting, object attributes, table properties, background color, and robot configuration. In addition, both intrinsic and extrinsic camera parameters were randomized to simulate different viewpoints and imaging conditions. The randomization schema is summarized in Table 3 in the Appendix. This procedure introduces controlled, multi-level variability, including lighting, materials, geometry, and sensing, which ensures the simulated dataset captures diverse visual and physical conditions for robust policy training and evaluation. We randomly sampled variations to generate 1,000 manipulation trials for each task. The complete data collection pipeline has been released as open-source, *allowing practitioners to flexibly modify and extend the framework according to their specific research needs.*

In addition, since we provide realistic data, practitioners can leverage it to guide video generation models [27] through in-context learning or fine-tuning. The generated data can be further augmented or refined using scene augmentation techniques (e.g., [28, 19]) applied in either simulated or real-world settings. RoboSynChallenge offers flexible integration with such pipelines, facilitating large-scale synthesized data collection.

Confidential Ground Truth and Leakage Prevention. Our evaluation protocol measures the model's real-world performance using diverse metrics (Section 1.4). This evaluation does not require access to the testing data or any ground-truth labels for verification. Instead, it assesses outcomes directly in the real world. Moreover, the training data consist only of successful trajectories collected in the learning environment. To quantify generalizability, the evaluation employs a held-out testing protocol in which the test environments are out-of-distribution relative to the training observations, ensuring minimal overlap and reducing the risk of dataset leakage.

All training and evaluation datasets will be made freely available to registered competition participants through online platforms (e.g., Hugging Face). The dataset is released under a **Creative Commons Attribution–NonCommercial–ShareAlike 4.0 (CC BY-NC-SA 4.0)** license, permitting research use while preventing commercial redistribution. Each release will include *metadata, documentation, and generation code* (for simulated data), in accordance with the NeurIPS Code of Ethics.

1.3 Tasks and application scenarios

Hardware. As Figure 2 shows, the evaluation of manipulation policy relies on a dual-arm platform built from two AgileX Piper manipulators. Each Piper is a 6-DoF research arm featuring open interfaces, designed for embodied manipulation and rapid system integration. This platform has been widely adopted in both prior competitions (e.g., RoboTwin [14], WBCD 2025 [3], and WBCD 2026 [4]) and recent research works [7, 30, 22, 31].

Manipulation tasks. Under the above hardware system, our design tasks are diverse and involve handling rigid objects, articulated objects, deformable objects, and tools. All tasks are performed based solely on visual observations and the robot's proprioceptive feedback. An overview of the real-world setup is shown in Figure 2. The workspace consists of an adjustable-height table with replaceable tablecloth textures and configurable lighting. We categorize the tasks as follows:

[3] https://wbcdcompetition.github.io/2025/index.html
[4] https://wbcdcompetition.github.io

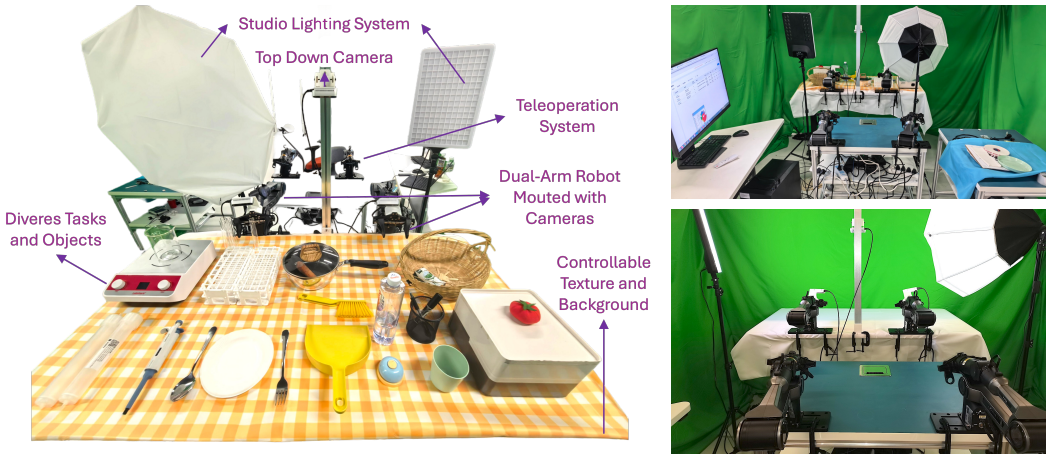

Figure 2: Real-world manipulation setup: The physical platform consists of a dual-arm robot, cameras, lighting, and various objects, enabling evaluation of all tasks included in the benchmark. To facilitate multiple rounds of real-world testing in a reliable and efficient manner, we have prepared *three sets of backup workstations*, each configured identically to the primary setup (Figure 4).

1) Entry-level tasks primarily include short-horizon, low-contact-complexity routines with clear affordances. As shown in Figure 3 (green background) and Table 4, they cover table rearrangement, click-bell, water pouring, and handle basket. These tasks emphasize stable perception, reliable grasp execution, and consistent motion trajectories under minimal environmental uncertainty. The focus lies in ensuring repeatable performance of simple action primitives rather than complex planning or intricate contact manipulation.

2) Mid-level tasks consist of compound, sequential interactions requiring moderate coordination, adaptive control, and spatial reasoning. As described in Figure 3 (blue background) and Table 4, these include items hand-over, drawer open-and-place, and mixer operating. In contrast to entry-level routines, mid-level tasks introduce variable contact conditions, multi-object dependencies, and partial planning horizons. Evaluation focuses on assessing smooth transition across motion phases, consistent force regulation, and robustness to mild external perturbations.

3) High-level tasks comprise fine-grained manipulation and domain-specific operations demanding high precision, multi-stage sequencing, and dynamic adaptation. As outlined in Figure 3 (red background) and Table 4, they include item assembly, pipette manipulation, and sample loading. These tasks require complex reasoning over ordered procedural steps, precise end-effector control, and environmental feedback integration. Benchmarking evaluates success through accuracy of placement or alignment, procedural consistency, and recovery capability under minor execution deviations.

Across these three levels, RoboSynChallengeprovides in-distribution testing (with respect to real-world training data) for model development and out-of-distribution testing, acting as a held-out benchmark to evaluate generalization quality throughout the competition.

1.4 Metrics

In our experiments, we present the evaluation protocol and metrics as follows.

Evaluation Protocol. We evaluate dexterous manipulation performance across multiple tasks under the following real-world variations: 1) *Background variation*: three table textures (wood, blue fabric, yellow grid). 2) *Lighting variation*: three distinct light positions with varying illumination colors. 3) *Object variation*: both *seen* and *unseen* object instances within the same task category, evaluating generalization to novel appearances. 4) *Distractor presence*: task-irrelevant objects placed in the scene at three difficulty levels (2, 4, and 8 distractors), sampled from a separate object set distinct from task objects. 5) *Spatial generalization*: evaluation on unseen positions within a predefined 3×3 grid. For each task, we first evaluate the policy under a canonical base configuration. We then systematically vary one factor at a time while keeping all other factors fixed. For each configuration,

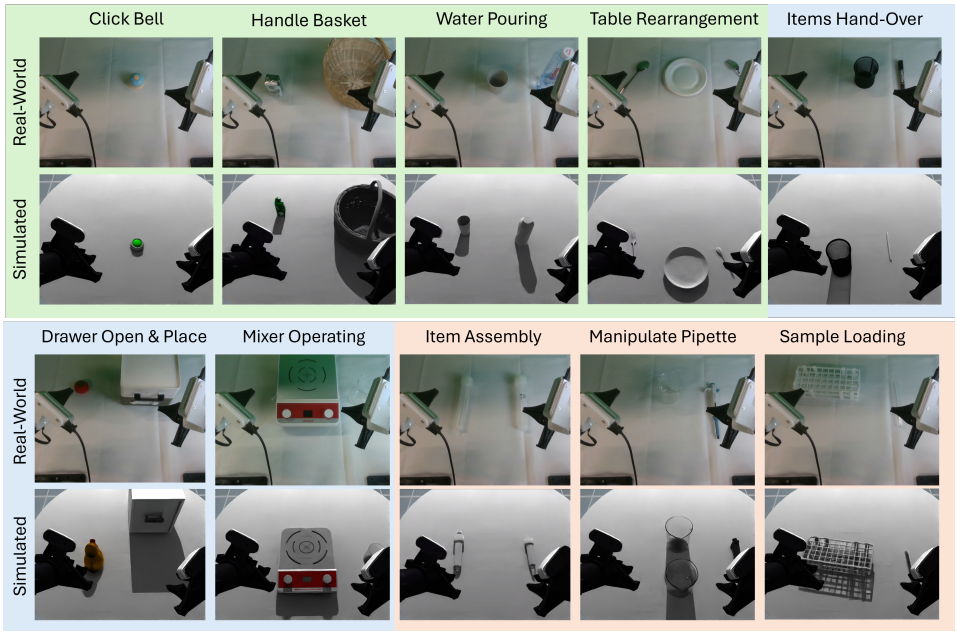

Figure 3: Visualizations of the realistic environments (upper row) and their corresponding simulated environments (lower row) across the three task categories: entry-level (green background), mid-level (blue background), and high-level (red background).

we evaluate the task under multiple object positions by applying small random shifts, and report the resulting success rate across these variations.

Evaluation Metrics. Based on the aforementioned evaluation protocol, we evaluate the performance of a dexterous policy according to the following methods: *1) Success Rate (SR)* (in percentage) measures how often a dexterous manipulation policy completes a task according to predefined success criteria (see Table 4). 2) *Inference Time* measures how quickly the policy produces actionable commands during inference; lighter models typically run faster. All models are deployed on the same machine with an NVIDIA A800 GPU to ensure fair comparison. 3) *Action Steps* count the number of control steps the model takes to finish a task, such that a higher count can indicate added latency due to error detection, feedback, and resolution, or reflect unresolved failures where the task remains incomplete, and the action budget is exhausted.

1.5 Baselines, code, and material provided

Baselines. The baselines for our task include five representative policy families, covering sequence modeling, generative action prediction, vision-language-action policies, and world-action modeling: 1) **Action Chunking Transformer (ACT) [3]**: A Transformer-based imitation learning policy that predicts temporally extended chunks of future actions from visual observations and proprioceptive states. By generating action chunks rather than single-step commands, ACT reduces compounding errors, improves temporal consistency, and provides a strong lightweight baseline for bimanual manipulation under limited demonstrations. 2) **Diffusion Policy [4]**: A visuomotor policy that formulates action prediction as a conditional denoising diffusion process. This baseline is well suited for multimodal manipulation behaviors, since it can represent diverse feasible action trajectories conditioned on the current observation history and iteratively refine them into smooth executable controls. 3) $\boldsymbol{\pi_0}$ [32]: A VLA model that combines a flow-matching architecture with a pre-trained vision-language model (VLM). The flow-matching component enables smooth trajectory learning between observations and actions, while the VLM provides strong visual-semantic grounding. 4) $\boldsymbol{\pi_{0.5}}$ [33]: A VLA model that builds upon π_0 by incorporating partial fine-tuning of the pre-trained vision-language backbone within the flow-matching framework. This allows $\pi_{0.5}$ to adapt more effectively to downstream tasks while retaining strong generalization from pre-training. 5) **Motus [7]**: A unified latent action world model that integrates vision, language, and action understanding through

a mixture-of-transformers architecture. By leveraging optical-flow-based latent actions and large-scale multimodal data, Motus achieves strong generalization and state-of-the-art performance across both simulated and real-world manipulation tasks. Together, these baselines establish a progressively richer comparison suite: ACT and Diffusion Policy represent widely used visuomotor imitation-learning methods, π_0 and $\pi_{0.5}$ evaluate recent VLA-style generalist policies, and Motus tests world-action-model-based reasoning over latent actions. For each baseline, we will provide standardized data loaders, training scripts, checkpoint formats, and evaluation wrappers so participants can reproduce results and compare new methods under the same Sim2Real protocol. The initial experimental results for the currently evaluated VLA and WAM baselines, trained using simulation-only or real-only data, are summarized in Table 5 in Appendix B below. Furthermore, we plan to incorporate a broader range of state-of-the-art baselines in future work, including VLA models such as **RDT-1B** [34] and WAMs like **DreamZero** [9].

Protocol of Releasing Code and Data We plan to release the starting kit through a dedicated GitHub repository that will contain all necessary materials for participants to easily join the competition. The repository will include: 1) Baseline code for model training and evaluation, 2) Data-loading and preprocessing tools for both simulated and real-world datasets, and 3) Sample simulated data together with a data-generation pipeline to enable participants to reproduce and extend training environments. In addition, we will release the real-world dataset and the complete simulated dataset (with data generator) on a public hosting platform (Hugging Face) to ensure accessibility and scalability.

1.6 Website, tutorial, and documentation

A dedicated competition website[5] will serve as the central hub for all information related to the event. The website will comprehensively present the competition overview, detailed timeline, participation steps, and relevant submission instructions. It will feature a FAQ/Tutorial section offering step-by-step guidance on installing the simulation environment, generating datasets, and testing models in both simulated and real-world settings. To ensure open communication, participants will be able to contact the organizers directly via robosynchallenge@gmail.com, with responses and updates maintained regularly. All content, including documentation and tutorials, will be made available well in advance of the competition start date, and the website will go live within two weeks of acceptance notification. The official GitHub repository [6] will host the codebase, baseline models, and data tools, while an accompanying white paper will describe the competition design, problem formulation, and technical foundation supporting the challenge.

References

[1] Christian Smith, Yiannis Karayiannidis, Lazaros Nalpantidis, Xavi Gratal, Peng Qi, Dimos V. Dimarogonas, and Danica Kragic. Dual arm manipulation - A survey. Robotics and Autonomous Systems, 60(10):1340–1353, 2012.

[2] Oliver Kroemer, Scott Niekum, and George Konidaris. A review of robot learning for manipulation: Challenges, representations, and algorithms. Journal of Machine Learning Research, 22(30):1–82, 2021.

[3] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In Robotics: Science and Systems, RSS, 2023.

[4] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In Robotics: Science and Systems, RSS, 2023.

[5] Yueen Ma, Zixing Song, Yuzheng Zhuang, Jianye Hao, and Irwin King. A survey on vision-language-action models for embodied ai. arXiv preprint arXiv:2405.14093, 2024.

[6] Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. arXiv preprint arXiv:2504.02792, 2025.

[5] Competition website: `https://robosyn-bench.net/`

[6] GitHub repository: `https://github.com/EDEM-AI/RoboSynChallenge`

[7] Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, et al. Motus: A unified latent action world model. arXiv preprint arXiv:2512.13030, 2025.

[8] Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, et al. Causal world modeling for robot control. arXiv preprint arXiv:2601.21998, 2026.

[9] Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, et al. World action models are zero-shot policies. arXiv preprint arXiv:2602.15922, 2026.

[10] Ying Zheng, Lei Yao, Yuejiao Su, Yi Zhang, Yi Wang, Sicheng Zhao, Yiyi Zhang, and Lap-Pui Chau. A survey of embodied learning for object-centric robotic manipulation. Machine Intelligence Research, pages 1–39, 2025.

[11] Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J Davison. Rlbench: The robot learning benchmark & learning environment. IEEE Robotics and Automation Letters, 5(2):3019–3026, 2020.

[12] Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. IEEE Robotics and Automation Letters (RA-L), 7(3):7327–7334, 2022.

[13] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. arXiv preprint arXiv:2306.03310, 2023.

[14] Yao Mu, Tianxing Chen, Shijia Peng, Zanxin Chen, Zeyu Gao, Yude Zou, Lunkai Lin, Zhiqiang Xie, and Ping Luo. Robotwin: Dual-arm robot benchmark with generative digital twins (early version). arXiv preprint arXiv:2409.02920, 2024.

[15] Adina Yakefu, Bin Xie, Chongyang Xu, Enwen Zhang, Erjin Zhou, Fan Jia, Haitao Yang, Haoqiang Fan, Haowei Zhang, Hongyang Peng, et al. Robochallenge: Large-scale real-robot evaluation of embodied policies. arXiv preprint arXiv:2510.17950, 2025.

[16] Yash Jangir, Yidi Zhang, Kashu Yamazaki, Chenyu Zhang, Kuan-Hsun Tu, Tsung-Wei Ke, Lei Ke, Yonatan Bisk, and Katerina Fragkiadaki. Robotarena ∞: Unlimited robot benchmarking via real-to-sim translation. In International Conference on Learning Representations, ICLR, 2026.

[17] Pranav Atreya, Karl Pertsch, Tony Lee, Moo Jin Kim, Arhan Jain, Artur Kuramshin, Clemens Eppner, Cyrus Neary, Edward Hu, Fabio Ramos, et al. Roboarena: Distributed real-world evaluation of generalist robot policies. In Proceedings of the Conference on Robot Learning (CoRL 2025), 2025.

[18] Yiting Chen, Kenneth Kimble, Edward H Adelson, Tamim Asfour, Podshara Chanrungmaneekul, Sachin Chitta, Yash Chitambar, Ziyang Chen, Ken Goldberg, Danica Kragic, et al. Manipulationnet: An infrastructure for benchmarking real-world robot manipulation with physical skill challenges and embodied multimodal reasoning. arXiv preprint arXiv:2603.04363, 2026.

[19] Ajay Mandlekar, Soroush Nasiriany, Bowen Wen, Iretiayo Akinola, Yashraj Narang, Linxi Fan, Yuke Zhu, and Dieter Fox. Mimicgen: A data generation system for scalable robot learning using human demonstrations. In Annual Conference on Robot Learning, CoRL, 2023.

[20] Shengliang Deng, Mi Yan, Songlin Wei, Haixin Ma, Yuxin Yang, Jiayi Chen, Zhiqi Zhang, Taoyu Yang, Xuheng Zhang, Wenhao Zhang, et al. Graspvla: a grasping foundation model pre-trained on billion-scale synthetic action data. arXiv preprint arXiv:2505.03233, 2025.

[21] Guiliang Liu, Yueci Deng, Runyi Zhao, Huayi Zhou, Jian Chen, Jietao Chen, Ruiyan Xu, Yunxin Tai, and Kui Jia. Dexscale: Automating data scaling for sim2real generalizable robot control. In International Conference on Machine Learning, ICML, 2025.

[22] Runyi Zhao, Sheng Xu, Ruixing Jin, Yueci Deng, Yunxin Tai, Kui Jia, and Guiliang Liu. Sim2real VLA: Zero-shot generalization of synthesized skills to realistic manipulation. In International Conference on Learning Representations, ICLR, 2026.

[23] Muhayy ud Din, Waseem Akram, Lyes Saad Saoud, Jan Rosell, and Irfan Hussain. Vision language action models in robotic manipulation: A systematic review. ArXiv, abs/2507.10672, 2025.

[24] Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. arXiv preprint arXiv:2406.02523, 2024.

[25] EmbodiChain Developers. Embodichain: An end-to-end, gpu-accelerated, and modular platform for building generalized embodied intelligence., November 2025.

[26] Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, et al. World simulation with video foundation models for physical ai. arXiv preprint arXiv:2511.00062, 2025.

[27] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314, 2025.

[28] Zhengrong Xue, Shuying Deng, Zhenyang Chen, Yixuan Wang, Zhecheng Yuan, and Huazhe Xu. Demogen: Synthetic demonstration generation for data-efficient visuomotor policy learning. arXiv preprint arXiv:2502.16932, 2025.

[29] Tianxing Chen, Kaixuan Wang, Zhaohui Yang, Yuhao Zhang, Zanxin Chen, Baijun Chen, Wanxi Dong, Ziyuan Liu, Dong Chen, Tianshuo Yang, et al. Benchmarking generalizable bimanual manipulation: Robotwin dual-arm collaboration challenge at cvpr 2025 meis workshop. arXiv preprint arXiv:2506.23351, 2025.

[30] Yi-Lin Wei, Haoran Liao, Yuhao Lin, Pengyue Wang, Zhizhao Liang, Guiliang Liu, and Wei-Shi Zheng. Cyclemanip: Enabling cyclic task manipulation via effective historical perception and understanding. arXiv preprint arXiv:2512.01022, 2025.

[31] Ruixiang Wang, Qingming Liu, Yueci Deng, Guiliang Liu, Zhen Liu, and Kui Jia. Eva: Aligning video world models with executable robot actions via inverse dynamics rewards. 2026.

[32] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. arXiv preprint arXiv:2410.24164, 2024.

[33] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. arXiv preprint arXiv:2504.16054, 2025.

[34] Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. RDT-1b: a diffusion foundation model for bimanual manipulation. In International Conference on Learning Representations, ICLR, 2025.

A Technical Details

A.1 Data Collection Protocol in the Real World

To evaluate real-world generalization, each manipulation task was conducted under five experimental conditions (varying background, lighting, and distractors, seen in Table 2), combined with four positions and three orientations. This systematic setup yields 60 diverse samples per task to capture varied spatial and visual factors.

Table 2: Summary of real-world data collection conditions.

Background	Lighting Setting	Additionals	Description
White	Fixed lighting	None	Standard setting with neutral background and lighting.
White	Enhanced lighting	None	Increased illumination to vary lighting conditions.
White	Fixed lighting	2–3 distractors	Includes additional objects to test robustness.
Blue	Fixed lighting	None	Alternative background color to introduce visual contrast.
Yellow	Fixed lighting	None	Textured background used to assess visual generalization.

A.2 Data Generation with EmbodiChain

EmbodiChain [25] is an end-to-end, GPU-accelerated, and modular platform for embodied AI research that integrates high-performance simulation, automated data pipelines, and flexible learning tools, thereby supporting rapid experimentation and effective Sim2Real transfer. It consists of three tightly coupled stages for scalable, diverse, and physically-grounded dataset generation.

1) Generative Simulation of Learning Environment. To overcome the limited diversity of manually designed simulation environments, EmbodiChain employs a two-stage generative framework. First, it synthesizes simulation-ready assets via generative models followed by multi-objective optimization to ensure geometric fidelity, physical plausibility, and simulator compatibility. Second, these assets are composed into fully functional scenes through gradient-based layout synthesis that optimizes object placement for physical realism and robot reachability. This process yields physically consistent, richly annotated environments that form the foundation of large-scale embodied data generation.

2) Data Scaling via Domain Expansion. Building on the generated environments, EmbodiChain scales embodied data by automatically generating and expanding robot interaction trajectories to improve coverage and robustness. It promotes functional diversity through reachability-aware sampling that selects kinematically feasible robot states maximizing task-space dissimilarity (e.g., in end-effector approach direction, contact geometry, and interaction outcomes), reducing trajectory homogenization common in teleoperation and conventional planners. To further strengthen robustness, a closed-loop error recovery module detects execution failures (e.g., slippage, misaligned grasps, boundary violations) and reactively replans corrective motions, which are relabeled and reintegrated as supervision for recovery behaviors.

3) Sim2Real Generalization via Online Data Streaming. To facilitate scalable Sim2Real transfer, EmbodiChain implements an *Online Data Streaming (ODS)* mechanism that continuously feeds diverse experiences from simulation into the learning loop. A streaming-based visual augmentation module perturbs lighting, textures, and sensor parameters on the fly, enriching perceptual diversity and mitigating overfitting to simulation-specific appearances. In parallel, an asynchronous shared-memory architecture enables real-time data exchange between simulation and learning processes through lock-free circular buffers, maximizing experience throughput and sample efficiency.

Table 3: Summary of simulated data randomization parameters.

Feature	Parameter	Description
Light	intensity_range: [*min*, *max*]	Lighting intensity variation.
	position_range: [[x_{min}, y_{min}, z_{min}], [x_{max}, y_{max}, z_{max}]]	Light source position randomization.
	color_range: [[0.6, 0.6, 0.6], [1.0, 1.0, 1.0]]	Light color variation.
Object location	init_pos: [x, y, z]	Object initial position.
	position_range: [[$-x$, $-y$, $-z$], [$+x$, $+y$, $+z$]]	Offset range from initial position.
Object orientation	rotation_range: [*min*, *max*]	Initial object rotation randomization.
Object material	base_color_range: [[0.2, 0.2, 0.2], [1.0, 1.0, 1.0]]	Object surface color variability.
Object size	scale: [x, y, z]	Uniform or axis-specific scaling.
Table	position_range: [[0, 0, -0.04], [0, 0, 0.04]]	Table height variation (±4 cm).
	random_texture_prob: p_{texture}	Probability of random table texture.
	base_color_range: [[r_{min}, g_{min}, b_{min}], [r_{max}, g_{max}, b_{max}]]	Table color range.
Background color	base_color_range: [[0.2, 0.2, 0.2], [1.0, 1.0, 1.0]]	Background plate color variation.
Camera (intrinsics)	$f_{x,\text{range}}$	Focal length range along x-axis.
	$f_{y,\text{range}}$	Focal length range along y-axis.
Camera (extrinsics)	pos_range: [[0, -0.02, 0], [0.02, 0.02, 0.01]]	Random camera position (XYZ).
	euler_range: [[-0.175, -0.175, -0.175], [0.175, 0.175, 0.175]]	Camera rotation (roll–pitch–yaw).
Robot init.	eef_pos_range: [[-0.01, -0.01, -0.01], [0.01, 0.01, 0]]	End-effector position variation.
	$q_{\text{pos,range}}$: ([$j_{1,\text{min}}$,...],[$j_{1,\text{max}}$,...])	Joint configuration randomization.
Distractors	Common items (e.g., bowls, cups, toys)	Distractors for scene complexity.

Table 4: Summary of Task Descriptions, Steps, and Success Criteria

Task Description	Steps	Success Criteria
Click Bell	(1) Move the arm toward the bell. (2) Align and press the button. (3) Return the arm to the initial position.	The button has been successfully pressed within a limited number of action steps.
Items Hand-Over and Place	(1) Grasp the pen. (2) Pass the pen to another arm. (3) Place the pen in the brush pot.	The pen has been successfully placed in the brush pot within a limited number of action steps.
Dual-Arm Water Pouring	(1) Grasp the water bottle and the cup. (2) Pour the water from the water bottle into the cup. (3) Place both objects securely on the table.	The water has been successfully poured into the cup, and both items are placed on the table within a limited number of action steps.
Table Rearrangement	(1) Grasp the spoon and the fork. (2) Arrange them neatly on both sides of the plate.	The spoons and forks have been arranged around the plate within a limited number of action steps.
Basket Pick-and-Place	(1) Grasp the milk box. (2) Place the milk box in the basket. (3) Grasp the basket. (4) Place the basket in the center of the table.	The milk box has been placed into the basket, and the basket has been set at the table center within a limited number of action steps.
Drawer Open and Place	(1) Grasp the handle. (2) Open the drawer. (3) Grasp an item. (4) Place the item in the drawer. (5) Close the drawer.	The item has been successfully placed in the drawer within a limited number of action steps.
Mixer Operating	(1) Grasp the beaker. (2) Place the beaker on the mixer. (3) Start the mixer.	The beaker was placed on the mixer and stirring successfully started within a limited number of action steps.
Item Assembly	(1) Grasp the silicone tubes. (2) Adjust them to a horizontal position. (3) Move one tube forward. (4) Move the other tube and splice together.	The two silicone tubes have been successfully joined together within a limited number of action steps.
Manipulate Pipette	(1) Grasp the pipette. (2) Insert pipette into the first beaker. (3) Another arm presses the pipette button.	The pipette has been correctly inserted into the beaker and the button has been successfully pressed within a limited number of action steps.
Sample Loading	(1) Grasp the test tube. (2) Pass the test tube to another arm. (3) Place the test tube into the rack.	The test tube has been successfully placed in the rack within a limited number of action steps.

B Evaluation Results of baselines

This section presents the baseline evaluation results, which encompass performance comparisons across policies trained on pure simulation data, pure real-world data, as well as the π_0, $\pi_{0.5}$, and Motus model variants. The evaluations are conducted across three core dimensions: success rate, action steps, and inference time.

Crucially, the empirical results show that the metrics of models trained on our simulation data are closely comparable to, and in several scenarios even outperform, those trained on real-world data when deployed in the real world. This demonstrates that our simulation synthesized data has comparable quality with real data, establishing a robust foundation for participants to conduct subsequent Sim2Real policy training and model optimization.

Figure 4: The demonstration showcases our backup hardware, which shares the same robot configurations and setup. This ensures that real-world evaluations can be conducted reliably and at scale.

Table 5: Experimental Results of Baselines: Success Rate ($x/20$ or %), Action Steps (max 1000), and Real Time (s).

Model	Click Bell			Items Hand-Over and Place			Dual-Arm Water Pouring			Table Rearrangement		
	SR	Steps	Time	SR	Steps	Time	SR	Steps	Time	SR	Steps	Time
pi0 (sim)	8/20	625.30	**63.78**	5/20	834.75	83.60	6/20	898.60	89.95	7/20	788.20	79.07
pi0 (real)	5/20	860.45	86.05	5/20	844.00	86.10	4/20	917.70	92.69	8/20	742.70	74.27
pi0.5 (sim)	10/20	647.55	65.53	7/20	791.25	**80.00**	7/20	877.45	**87.90**	**12/20**	**612.90**	**61.31**
pi0.5 (real)	6/20	791.20	80.70	5/20	832.90	84.12	6/20	872.15	**87.22**	**12/20**	628.80	63.51
Motus (sim)	**13/20**	**463.30**	79.09	**10/20**	584.70	97.52	**8/20**	**667.30**	111.00	4/20	864.25	143.75
Motus (real)	**14/20**	**420.50**	70.11	**12/20**	**492.30**	83.65	6/20	744.95	125.97	3/20	900.05	153.78

Model	Basket Pick-and-Place			Drawer Open and Place			Mixer Operating			Item Assembly		
	SR	Steps	Time	SR	Steps	Time	SR	Steps	Time	SR	Steps	Time
pi0 (sim)	5/20	835.40	83.56	6/20	821.40	82.23	3/20	897.05	90.09	0/20	1000.00	102.07
pi0 (real)	6/20	796.70	80.47	8/20	757.70	77.29	1/20	967.55	98.69	0/20	1000.00	99.86
pi0.5 (sim)	**10/20**	663.25	**66.42**	11/20	664.40	**67.08**	**4/20**	**864.50**	**87.11**	0/20	1000.00	104.29
pi0.5 (real)	9/20	697.90	71.88	**12/20**	645.70	**65.22**	3/20	901.75	90.18	0/20	1000.00	101.02
Motus (sim)	**10/20**	827.95	132.29	10/20	593.15	102.70	2/20	936.25	154.80	0/20	1000.00	166.22
Motus (real)	9/20	**608.55**	101.27	11/20	**546.60**	93.96	0/20	1000.00	166.41	0/20	1000.00	167.37

Model	Manipulate Pipette			Sample Loading			Task Average		
	SR	Steps	Time	SR	Steps	Time	SR	Steps	Time
pi0 (sim)	2/20	960.65	96.29	2/20	946.85	94.73	22.00%	898.12	90.56
pi0 (real)	0/20	1000.00	108.46	3/20	920.35	92.04	22.50%	881.15	90.20
pi0.5 (sim)	2/20	953.15	**95.82**	**4/20**	**900.05**	**89.97**	**38.50%**	797.55	**80.55**
pi0.5 (real)	**4/20**	**898.75**	**91.67**	3/20	914.95	93.32	33.00%	821.65	82.35
Motus (sim)	**4/20**	905.35	149.33	2/20	945.70	157.82	31.50%	778.80	133.76
Motus (real)	0/20	1000.00	164.84	0/20	1000.00	166.85	27.50%	**721.35**	129.43