GAM: Generalized Action Model for Robotic Manipulation

Uday Kamal*†, Jeffrey Jiang*†, Chaitanya Mitash*†, Weiyao Wang†, Ikechukwu Obi-Okoye†, Che Wang†, Aatif Jiwani‡, Huitan Mao†, Hai Nguyen†, Michel Breyer†, Wenyu Xia†, Nikhil Mishra‡, Kostas Bekris†, Kapil Katyal†
†Amazon Robotics ‡Amazon FAR

***Abstract*—We present the Generalized Action Model (GAM), a production-grade foundation model that unifies robotic action generation across diverse tasks and embodiments through a vision-language-action (VLA) pipeline. GAM addresses two fundamental barriers in scaling robotic manipulation: the lack of a unified representation for diverse robot end-effectors and the prohibitive cost of acquiring high-quality interaction data at scale. Our approach introduces (1) a unified language-prompted policy and critic that generates and scores diverse manipulation actions—including suction grasps, pinch grasps, caging, and placements—from a single model, (2) a scalable offline data generation pipeline that recomputes dense action candidates and quality labels in simulation from real-world observations, and (3) an end-effector encoding that enables zero-shot transfer to unseen hardware. We validate GAM on a fleet of robotic work-cells, where it has executed over 10 million pick-and-place cycles with greater than 95% pick and greater than 90% place success rates. The same model generalizes to hybrid end-effectors with distinct grasping modes at greater than 90% success.**

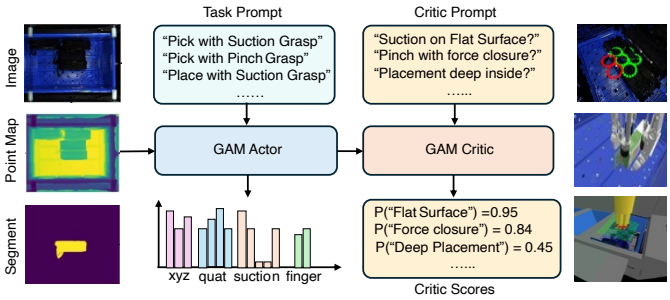

Fig. 1. Overview of the GAM pipeline. Scene observations are processed by GAM-Actor, which generates diverse candidate actions conditioned on task prompts. GAM-Critic scores each candidate against language-specified quality criteria to select the top actions for execution.

I. Introduction

General-purpose robotic manipulation has been constrained by fragmented, task-specific pipelines. In practice, systems for pinch-grasping [1], [2], suction-based picking [3], [4], and object placement [5], [6] are developed and maintained independently, each with their own data, models, and inference stacks. This fragmentation limits generalization and increases deployment overhead. Recent work has shifted toward cross-embodiment training to learn generalist Vision-Language-Action (VLA) policies. Open X-Embodiment [7] aggregates diverse trajectories from multiple institutions, enabling foundation models such as π_0 [8] and RT-2 [9] to map visual and semantic inputs to robotic control. Concurrently, diffusion-based policies [10]–[12] have demonstrated strong performance in capturing multi-modal action distributions for continuous control. However, these approaches face persistent challenges: data collection at scale remains a bottleneck due to reliance on teleoperation or online interaction, and genuine cross-embodiment transfer remains unsolved as most models encode actions in joint-space or use formulations tied to a specific robot morphology. Prior work on task-specific grasping [13], [14] and multi-task learning [15], [16] has not simultaneously addressed these challenges at production scale.

In this work, we present the **Generalized Action Model (GAM)**, a complete VLA pipeline from data generation to real-world deployment. GAM generates 6-DoF end-effector poses and actuation commands for pick-and-place tasks in cluttered environments. Our key contributions are:

- **Unified Prompted Policy and Critic:** A single Diffusion Transformer model generates diverse, multi-modal actions—suction, pinch, caging grasps, and placements—steered by natural language prompts. A companion language-prompted critic scores candidates along multiple quality dimensions, enabling deployment flexibility.
- **Scalable Data Generation:** An offline pipeline that takes real-world scene observations and recomputes dense action candidates and their quality scores in simulation, producing millions of samples for training the actor and critic without an extensive data collection effort.
- **Cross-Embodiment Generalization:** A tokenized end-effector encoding that enables zero-shot transfer to unseen hardware layouts. The same scenes can be re-labeled for new embodiments in simulation, eliminating the need for physical data collection when bringing up new grippers.
- **Real-World Validation at Scale:** Deployment across a fleet of robotic work-cells with over 10 million pick-and-place cycles across open-world object sets, achieving $>95\%$ pick and $>90\%$ place success from a single unified model, with $>90\%$ success on hybrid grippers across multiple grasping modes.

*Equal contribution.

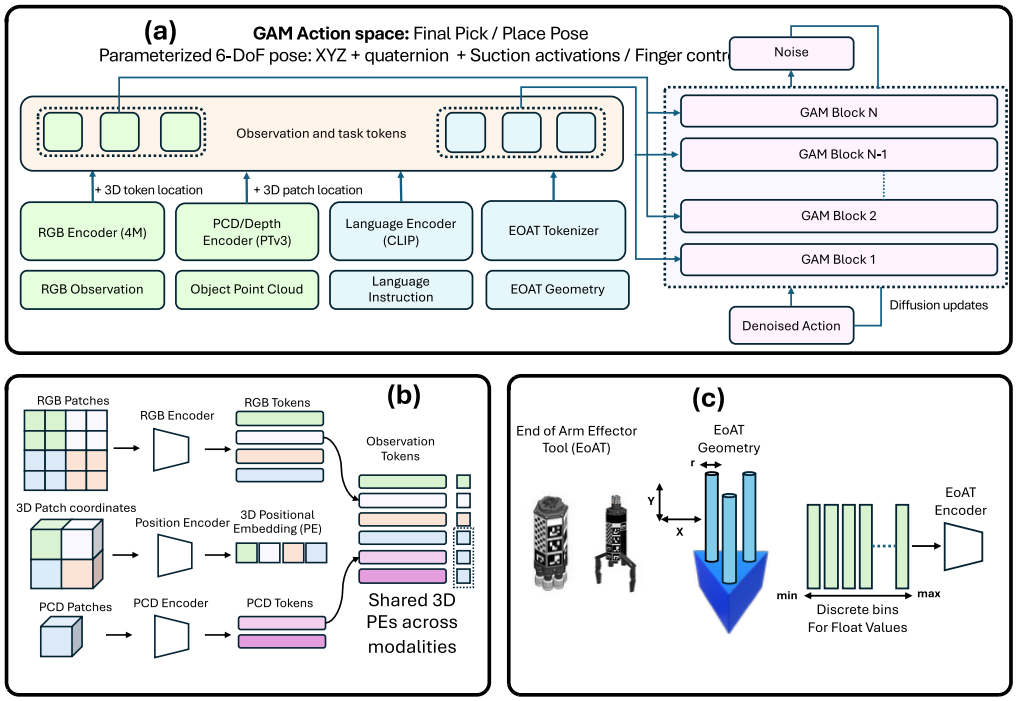

Fig. 2. GAM architecture. (a) The DiT backbone processes multimodal tokens—from RGB images, 3D point clouds, end-of-arm tool geometry, and language task specifications—through an alternate conditioning strategy to generate 6-DoF end-effector poses and actuation commands via iterative denoising. (b) Spatial association across RGB and point cloud modalities via shared 3D positional embeddings. (c) End-of-arm tool encoding: continuous geometric parameters are discretized into fixed bins and mapped to learnable embeddings.

II. Approach

An overview of the GAM pipeline is shown in Fig. 1. Given a scene observation (RGB image, depth map, and instance segmentation), GAM-Actor generates a diverse set of candidate actions conditioned on a natural language task prompt, producing 6-DoF end-effector poses and actuation parameters (e.g., suction activations, finger width). GAM-Critic then scores each candidate against multiple quality criteria—also specified via language prompts—to select the best actions for downstream execution.

A. Unified Prompted Policy and Critic

GAM-Actor is a generative policy parameterized by a Diffusion Transformer (DiT) backbone [10], [11] that models the conditional distribution of continuous actions through iterative denoising. Unlike prior diffusion policies that operate in joint-space or are conditioned on a fixed embodiment, GAM generates actions in task-space (6-DoF end-effector poses) conditioned on the scene, end-effector geometry, and a natural language task prompt.

Generating Diverse Multi-Modal Actions: The key design goal is a single model that can produce qualitatively different manipulation behaviors—suction grasps, pinch grasps, post-pick caging, and placements—depending on the task prompt. For suction grasping, the model predicts the end-effector pose and per-cup activation signals. For finger-based grasping, it additionally predicts finger width and depth. For placement, it conditions on the selected pick action and the destination scene to generate placement poses. The diffusion backbone naturally captures the multi-modality of these action distributions: for a given scene, there are many valid grasps with varying approach angles, suction patterns, and finger configurations, and the model learns to cover this distribution rather than collapse to a single mode.

Architecture: The DiT backbone (Fig. 2a) processes multimodal tokens from dedicated encoders: a ViT-based encoder (4M [17]) for RGB images, Point Transformer V3 [18] for object-centric 3D geometry, CLIP for language task prompts, and a tokenized end-effector representation (detailed in Section II-C). RGB and point cloud modalities are spatially grounded via shared sinusoidal 3D positional embeddings (Fig. 2b), enabling explicit cross-modal spatial association. We employ *alternate conditioning* [11], which reduces per-block attention cost and introduces an inductive bias that separates spatial reasoning from task-level intent: even-indexed blocks attend to spatial tokens (RGB + point cloud) for scene grounding, while odd-indexed blocks attend to task-embodiment tokens (language + end-effector) for intent and kinematic alignment. The model is trained to directly predict the clean action $\hat{a}_0$ from noisy input a_t, minimizing $\mathcal{L} = \mathbb{E}\left[\|a_0 - f_\theta(a_t, t, C)\|_2^2\right]$.

Language-Prompted Critic for Flexible Inference: GAM-Critic is a learned scoring function trained alongside the actor. Rather than predicting a single scalar quality score, the critic evaluates each candidate action against *multiple* quality criteria, each specified via a distinct language prompt—

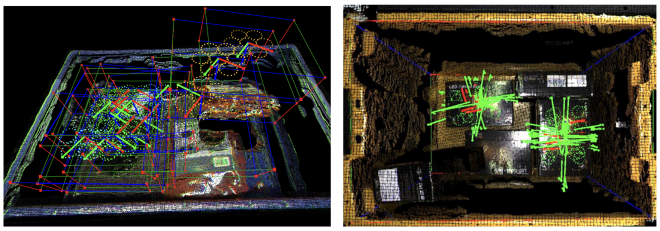

Fig. 3. Dense action labels generated from real-world scenes - placement and pinch grasping are shown. Each scene yields thousands of candidate actions with simulation-derived quality scores for training both the actor and critic.

for example, "probability of collision", "suction seal quality", "force closure for pinch", or "placement depth". This language-conditioned design makes the scoring flexible: new criteria can be added without architectural changes, and the relative importance of criteria can be adjusted at inference time depending on the task context.

At inference, the actor samples a configurable number of candidates (e.g., 64–100 per scene) from the generative distribution. The critic then scores all candidates, and the top-K collision-free actions are passed to the motion planner. Generating more candidates increases the likelihood of finding high-quality, reachable actions. At inference, we use a DPM++ scheduler [19] to reduce denoising from 1000 to 10 steps.

B. Scalable Data Generation Pipeline

A central challenge in training generalist manipulation policies is acquiring diverse, high-quality data. Methods based on teleoperation [8], [9] or online interaction [13] are expensive to scale and tightly coupled to specific hardware. Open X-Embodiment [7] aggregates trajectories across institutions, but the resulting data is heterogeneous in quality and sparsely covers any individual task domain. Our approach decouples data generation from physical robot execution through a fully offline pipeline that takes real-world observations and recomputes dense action labels in simulation.

Dense Action Sampling and Simulation-Based Scoring: The dataset curation pipeline ingests scenes (RGB images, depth maps) from large-scale datasets [20], [21] across varying conditions for pick-and-place manipulation. The key insight is that a single recorded scene can yield *thousands* of labeled training samples. For each scene, we sample a dense set of candidate actions in simulation using the captured 3D scene reconstruction. For grasping, specialized samplers generate thousands of candidate picks across the scene. Each candidate is then evaluated in simulation to produce quality labels: binary scores (e.g., collision-free, single-object contact) and continuous scores (e.g., surface flatness for suction seal, proximity to object center, expected number of active suction cups). For placement, we select the top collision-free grasps and simulate hundreds of plausible placements per grasp, scoring each for volume utilization, overflow risk, and collision status. These simulation-derived scores serve as ground-truth labels for training both the actor and the critic—the actor learns from the highest-quality actions, while the critic learns to predict the full spectrum of quality scores.

Fig. 4. (left) pick prediction showing activated suctions in green. (right) multi-modal grasps showing diversity in approach angles and suction combinations predicted on the same object

Training the Actor and Critic: GAM-Actor is supervised on the top 10% of actions per scene, ranked by combined quality scores, learning to generate actions that match the best demonstrations. GAM-Critic is trained on balanced positive and negative samples across all quality dimensions, learning to predict each score independently. This means the critic sees the full diversity of action quality within each scene—from excellent grasps to poor ones—enabling fine-grained discrimination at inference time. Fig. 3 illustrates the dense labels generated across different grasping modalities.

This approach offers several advantages: it provides orders of magnitude more training signal per scene than trajectory-based collection, it produces multi-dimensional quality labels enabling the critic to learn diverse scores, and it is inherently scalable—adding new scenes requires no additional robot time.

C. Cross-Embodiment Generalization

End-of-Arm Tool (EoAT) Encoding: To enable a single policy to generalize across diverse hardware—from single-suction cups to multi-cup arrays to finger-based grippers—we represent the end-effector as K cylindrical primitives, each parameterized by its 2D planar location (x_k, y_k) and radius r_k. These continuous parameters are discretized into fixed bins and mapped to learnable embeddings: $z_k^{eoat} = \text{Emb}_x(q(x_k)) + \text{Emb}_y(q(y_k)) + \text{Emb}_r(q(r_k))$. This creates a hardware-agnostic token vocabulary that captures the spatial layout and scale of any tool configuration (Fig. 2c). Compared to prior approaches that train separate models per end-effector or encode embodiment implicitly, our explicit geometric tokenization enables zero-shot transfer to unseen layouts.

Scene Re-Labeling for New Embodiments: A key benefit of simulation-based scoring is that existing scenes can be re-labeled for new end-effectors without physical data collection. For example, when bringing up a new hybrid EoAT (suction and fingers)—we apply the target end-effector geometry to existing scenes and recompute actions and quality scores with embodiment-specific criteria (e.g., pinch alignment, expected pinch force). This dramatically reduces the effort required for new hardware: the same real-world observations serve as the foundation, and only the simulation-based labeling changes.

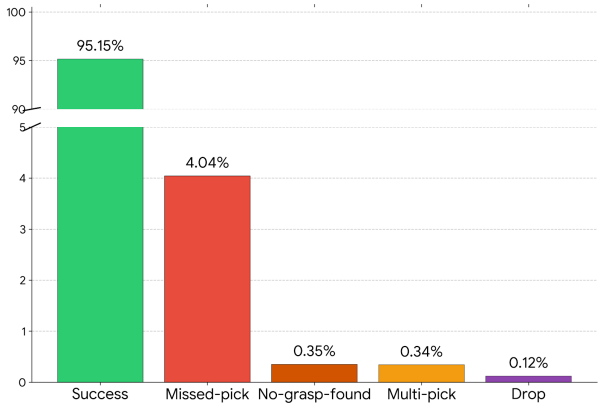

Fig. 5. Pick performance for suction-based picking evaluated over 10 million picks in real-world fulfillment setting with heterogeneous clutter of items. Missed-pick: pick was attempted but failed due to lack of seal, often on items with material or geometry that are hard for suction grasping; No-grasp-found: No collision-free, valid grasp was sampled on the object; Multi-pick: Picks up more than one object unintentionally; Drop: Drops the item during motion.

D. Real-World Validation at Scale

GAM is evaluated across a fleet of robotic work-cells performing high-throughput bin-to-bin item sorting, where a robotic arm picks items from source containers and places them into designated target containers under cluttered, real-world conditions. The same model handles both pick and place action generation, performing over 100,000 operations per day. This evaluation uses an end-effector with suction cups mounted over seven independently actuated pistons. Additional experiments are conducted with a hybrid gripper featuring three suction cups and two fingers, enabling a larger action space. While this experiment operates at a smaller scale with approximately 4,000 cycles, it runs in completely uncontrolled environments with arbitrary object clutter.

Suction-based grasping: given the sensing data, the GAM-Actor samples up to 100 grasps across the scene. The model directly predicts the six-degree-of-freedom pose of the gripper and suction activation signals across all suction cups. These grasps are then scored using the GAM-critic, which learns scores that maximize suction seal and the number of active suction cups while minimizing interference forces from other items. The top-scored collision-free grasp is executed. Evaluated over more than 10 million picks, this approach achieves greater than 95% success rate, where success is defined as the item being successfully picked from the tote and remaining stable within the robotic arm during fast motion. Figure 4 shows some qualitative results and Figure 5 plots the distribution of the remaining 5% failure cases.

Placement in clutter: during placement inference, the GAM-Actor takes images of the source and destination containers, along with the selected pick action, to produce the end-effector pose for placing the item into the cluttered destination container. A total of 64 placements are sampled across the destination container, and the top-scored placement is executed. Placement achieves greater than 90% success rate across 10 million cycles of placing into arbitrary clutter that often involves tight plcements as shown in Figure 7.

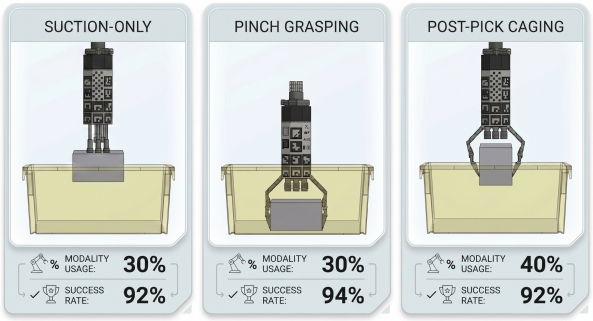

Fig. 6. Figure shows the execution and success rates for different grasp modalities across 4000+ cycles. While suction and pinch are self explanatory, post-pick caging refers to the action mode where the gripper uses suction to lift the item and then engages fingers to cage it. The model needs to sample grasp poses that allow caging along the correct orientation.

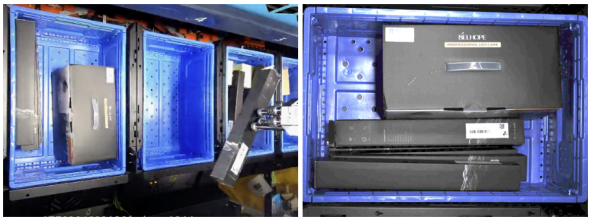

Fig. 7. Before / after images for tight placement. GAM generates final gripper pose based on the source, destination tote images and the selected grasp.

Hybrid, suction and finger based grasping: to evaluate application to new embodiments, we swapped the end-effector to a hybrid gripper. The GAM-Actor is prompted via language with the mode of grasp (suction, pinching, caging) to generate samples corresponding to that mode. For finger-based grasping, it additionally predicts the width and depth for the fingers along with the pose and suction activations. GAM-critic selects the best grasp to execute. This model was trained without any physical data collection on this embodiment. Figure 6 shows the success rates for different grasp modes over 4,000 executions in the real world.

III. Conclusion

We presented GAM, a complete VLA pipeline for robotic manipulation spanning offline data generation, multimodal policy learning, and production-scale evaluation. Our scalable offline data pipeline generates millions of densely labeled samples from historical scenes without teleoperation—addresses a key bottleneck in training generalist manipulation policies. The novel EoAT encoding and alternate conditioning strategy enable a single model to serve diverse tasks and embodiments. GAM achieves production-grade performance ($>95\%$ pick, $>90\%$ place) across a fleet of robots with over 10 million cycles, generalizes to unseen hardware via zero-shot transfer, and supports distinct grasping modes on hybrid end-effectors. Future work includes extending GAM to temporal and scene-level reasoning for multi-stage manipulation tasks across diverse embodiments.

REFERENCES

[1] H.-S. Fang, C. Wang, H. Fang, M. Gou, J. Liu, H. Yan, W. Liu, Y. Xie, and C. Lu, "AnyGrasp: Robust and efficient grasp perception in spatial and temporal domains," *IEEE Transactions on Robotics*, vol. 39, no. 5, pp. 3929–3945, 2023.

[2] J. Mahler and K. Goldberg, "Learning deep policies for robot bin picking by simulating robust grasping sequences," in *Proceedings of the 1st Annual Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 78, 2017, pp. 515–524.

[3] J. Mahler, M. Matl, X. Liu, A. Li, D. Gealy, and K. Goldberg, "Dex-Net 3.0: Computing robust vacuum suction grasp targets in point clouds using a new analytic model and deep learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 5620–5627.

[4] H. Cao, H.-S. Fang, W. Liu, and C. Lu, "SuctionNet-1Billion: A large-scale benchmark for suction grasping," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8718–8725, 2021.

[5] W. Liu, C. Paxton, T. Hermans, and D. Fox, "StructFormer: Learning spatial structure for language-guided semantic rearrangement of novel objects," in *2022 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 6322–6329.

[6] Y. Zhao, M. Bogdanovic, C. Luo, S. Tohme, K. Darvish, A. Aspuru-Guzik, F. Shkurti, and A. Garg, "AnyPlace: Learning generalizable object placement for robot manipulation," in *8th Annual Conference on Robot Learning (CoRL 2024)*, vol. 305. PMLR, 2025, pp. 4038–4057.

[7] Open X-Embodiment Collaboration *et al.*, "Open X-Embodiment: Robotic learning datasets and RT-X models," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 6892–6903.

[8] K. Black, N. Brown *et al.*, "π_0: A vision-language-action flow model for general robot control," *arXiv preprint arXiv:2410.24164*, 2024.

[9] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar *et al.*, "RT-2: Vision-language-action models transfer web knowledge to robotic control," in *7th Annual Conference on Robot Learning (CoRL)*, 2023.

[10] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," in *Robotics: Science and Systems (RSS)*, 2023.

[11] Y. Gu, Z. Wang, Y. Jiang, C. Qiu, L. Zhang, and J. Lin, "RDT-1B: A diffusion foundation model for bimanual manipulation," *arXiv preprint arXiv:2410.07864*, 2024.

[12] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 6840–6851.

[13] S. Levine, P. Pastor Sampedro, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 421–436, 2018.

[14] W. Yuan, A. Murali, A. Mousavian, and D. Fox, "M2T2: Multi-task masked transformer for object-centric pick and place," in *Proceedings of The 7th Conference on Robot Learning*, 2023, pp. 1–12.

[15] S. Reed, K. Zolna, E. Parisotto *et al.*, "A generalist agent," *Transactions on Machine Learning Research*, 2022.

[16] M. Shridhar, L. Manuelli, and D. Fox, "Perceiver-Actor: A multi-task transformer for robotic manipulation," in *Proceedings of The 6th Conference on Robot Learning*, 2022, pp. 785–799.

[17] D. Mizrahi, R. Bachmann, O. Kar, T. Yeo, M. Gao, A. Dehghan, and A. Zamir, "4M: Massively multimodal masked modeling," *Advances in Neural Information Processing Systems*, vol. 36, pp. 58 363–58 408, 2023.

[18] X. Wu, L. Jiang, P.-S. Wang, Z. Liu, X. Liu, Y. Qiao, W. Ouyang, T. He, and H. Zhao, "Point transformer V3: Simpler faster stronger," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 4840–4851.

[19] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu, "DPM-Solver++: Fast solver for guided sampling of diffusion probabilistic models," *Machine Intelligence Research*, pp. 1–22, 2025.

[20] C. Wang, J. van Baar, C. Mitash, S. Li, D. Randle, W. Wang, S. Sontakke, K. E. Bekris, and K. Katyal, "Demonstrating multi-suction item picking at scale via multi-modal learning of pick success," *arXiv preprint arXiv:2506.10359*, 2025, accepted to Robotics: Science and Systems (RSS 2025).

[21] C. Mitash, F. Wang, S. Lu, V. Terhuja, T. Garaas, F. Polido, and M. Nambi, "Armbench: An object-centric benchmark dataset for robotic manipulation," *arXiv preprint arXiv:2303.16382*, 2023.