Zero-Shot Generalization Capabilities of Cross-Embodiment Learning in a Simulated Environment

Hylke Westerdijk
Department of Artificial Intelligence
University of Groningen
Groningen, the Netherlands
h.p.westerdijk@student.rug.nl

Jort Hessel
Department of Artificial Intelligence
University of Groningen
Groningen, the Netherlands
j.hessel.5@student.rug.nl

***Abstract*—In recent years, cross-embodiment learning has emerged as a promising method to address data scarcity challenges in robotics, by enabling the training of a single policy across a variety of robot embodiments. This study replicates and extends the work of Doshi et al. [1], which demonstrated the potential for zero-shot generalization in cross-embodiment policies. We employ the CrossFormer model, a transformer-based architecture trained on the Open-X Embodiment (OXE) dataset, to evaluate its performance in zero-shot generalization on a grasping task using a simulated robotic arm. Our experimental setup included ten distinct objects and assessed performance using success rate, step count, gripper close count, and distance to the object to be grasped. The results suggest that, while the model achieved promising success rates for some objects, fine-tuning does seem to be necessary to fully leverage the model's capabilities. Grasp success appeared to be influenced by the robotic arm's visibility in the camera's field of view, domain gaps between the simulated and real-world environments, issues in the perception of spatial relations, and biases in training data.**

***Index Terms*—cross-embodiment, zero-shot, learning, generalization, robotics, CrossFormer, Open-X Embodiment (OXE)**

I. Introduction

In recent years, large Language Models (LLMs) have demonstrated remarkable success by leveraging vast amounts of data to improve performance across a diverse array of tasks. By utilizing extensive datasets scraped from a wide range of sources, these models achieve robustness and generalization. In robotics, however, obtaining similarly large datasets presents significant challenges. Different tasks bring different robot morphologies and different environments, and policies are usually trained to learn the features of these specific embodiments. Recently, however, cross-embodiment learning has emerged as a promising avenue to address this limitation.

For our purposes, we define cross-embodiment learning to be learning in which data is aggregated from various robot embodiments to train a single policy capable of executing different tasks across different robots. For instance, such a policy could be trained to grasp and manipulate objects regardless of whether the robot has one or two arms, exploiting commonalities in tasks and environments to enhance generalization. This approach holds the potential to advance robotics by enabling the aggregation of experimental data from research efforts worldwide, addressing the pervasive problem of data scarcity, and bypassing the need to learn the features of every new embodiment from scratch. Much like how LLMs transformed NLP, cross-embodiment learning could similarly catalyze advancements in robotics.

Yet, a critical question arises: how generalized is the knowledge these cross-embodiment policies can acquire? Would these models require fine-tuning for new tasks, or could they possibly even demonstrate zero-shot generalization? Doshi et al. [1] demonstrated that their cross-embodiment policy outperformed policies trained on single-robot datasets on a navigation task with a quadcopter embodiment, despite the cross-embodiment policy not being trained on quadcopter-specific data. In doing this, they demonstrated the interesting possibility of zero-shot generalization in cross-embodiment learning.

In this study, we aim to replicate Doshi et al.'s findings while conducting a qualitative analysis of the model's zero-shot performance. Through this exploration, we seek to uncover insights into the potential of cross-embodiment learning and identify new research questions to drive the field forward.

A. Related Work

Doshi et al. provided the foundational work on zero-shot generalization in cross-embodiment policies, demonstrating robust results in novel embodiments. In their study, they describe a method for training a single policy (the 'CrossFormer') across 20 robot embodiments with different observation and action spaces, including single-arm and dual-arm manipulators, quadrupeds, and ground-navigation robots. It processes observations (images, proprioception) and task specifications (language, goal images) through modality-specific tokenizers, feeding them into a shared transformer backbone. The model's output embeddings are directed to embodiment-specific action heads, predicting actions with the correct dimensionality for each robot. They trained with upweighted data from target robots (WidowX, ALOHA, Franka, LoCoBot, and Go1) while maintaining generalization. CrossFormer con-

sistently either matched or outperformed single-robot baselines without negative transfer. Their results suggest that training on sufficiently diverse data with a flexible architecture such as theirs does lead to effective cross-embodiment learning.

What these authors describe as potentially most related to their work, is a study by Yang et al. [2], which also focuses on the benefits of training a single policy on data from heterogeneous robots. They also demonstrated that co-training with navigation data can improve robustness and performance of goal-conditioned manipulation, and vice versa. To achieve this, they proposed a unified action space where actions have similar effects on the observations of different robots, despite variations in hardware (this is where their architecture differs from that of Doshi et al., since the latter avoids the need for manually aligning action spaces). Their model uses EfficientNet ConvNets for observation encoding and a diffusion policy for action output, which are combined with a transformer backbone. Their co-trained policy achieved a 20% higher success rate over a manipulation-only policy and a 5-7% improvement over a navigation-only policy on different robots. Their results equally suggest that data from very diverse sources can be usefully aggregated and used to co-train cross-embodiment policies.

To our knowledge, our study is the first to replicate and qualitatively analyze the zero-shot generalization from Doshi et al. on a robotic arm.

II. Methodology

A. CrossFormer

We employed the CrossFormer architecture proposed by Doshi et al. [1] for our experiments. This model was specifically designed to leverage cross-embodiment data for generalization across varied tasks. The CrossFormer first tokenizes camera views and/or proprioceptive information using a ResNet-26 encoder, and it tokenizes language instructions using FiLM. Then, it feeds the output of this tokenization process into the transformer backbone, which was co-trained with all embodiments. The transformer backbone has 8 attention heads, 12 layers, a token embedding size of 512, and an MLP dimension of 2048. The model has 130M parameters in total. The output of the transformer (an embedding sequence) is passed into one of the action heads, where it is projected to an embodiment-specific action space (single-arm cartesian position, navigation waypoints, bimanual joint positions, or quadruped joint positions; their architecture also allows for another action head to be added). We used the single-arm cartesian position output head, which predicts 4-step action sequences.

B. Dataset

The CrossFromer was trained on a subset of the Open-X Embodiment (OXE) dataset. The OXE dataset is a large-scale, open-source collection of robotic learning data meant to support research in cross-embodiment learning. It contains over 1 million real robot trajectories from 22 different robot embodiments, including single robot arms, bimanual robots, and quadrupeds. The data is gathered from 34 robotic research labs across 21 institutions. The dataset combines 60 existing robot datasets into a unified RLDS format, which is designed for efficient, parallelized use in deep learning frameworks. It features a variety of action spaces and input modalities, such as different numbers of RGB cameras, depth cameras, and point clouds. The Franka robot is the most common robot in the dataset, both in terms of the number of scenes and the total data volume. While the dataset focuses mainly on pick-and-place tasks, it also includes other tasks like wiping and assembly, with a wide range of objects from household items to food. The subset that CrossFormer was trained on consisted primarily of trajectories of single arm emboidments, but also contained significant amounts of trajectories from navigation, bimanual or quadruped embodiments.

C. Simulation environment

As no physical robotic arm was available, all experiments were conducted in a simulated environment. We evaluated the CrossFormer's performance on a Universal Robot (UR5e) robotic arm with a two-fingered Robotiq 2F-140 gripper in a PyBullet environment (code). The environment contains a table next to which the robotic arm is situated, and above which is an RGB-D camera looking down at the table and the objects potentially placed upon it. The objects used in the experiments are simulated Yale-CMU-Berkeley (YCB) objects, a standardized collection of household and everyday items, designed to benchmark robotic manipulation and grasping. It includes objects with different sizes, shapes, textures, and weights. We evaluated the model's performance only on the isolated scenario. This scenario is designed to evaluate the robot's ability to accurately identify and grasp a single object that was placed independently, and without interference from other objects within the workspace. The object's placement within the workspace was randomized, to ensure variability across runs and to test the robot's adaptability to different spatial positions.

D. Alignment

Initial trials revealed that the robotic arm was not visible within the camera's field of view, likely hindering the policy's ability to perform effectively. By mimicking configurations observed in the Bridge v2 dataset—where the arm is consistently visible—we repositioned the camera and arm to ensure alignment. Alignment of the simulation environment was performed iteratively, until performance improved significantly with the banana object.

Our simulation environment was tailored to a banana-shaped object, chosen for its elongated structure. This decision was informed by observations that objects with longer dimensions aligned better with the policy's capabilities. Notably, this setup introduced potential biases, prompting us to compare performance across ten distinct objects.

E. Experimental Setup

We conducted 10 runs per object across 10 distinct objects, focusing exclusively on grasping rather than manipulation.

Grasp quality was not evaluated; instead, success was defined as the completion of a grasp, including those deemed technically successful (see *F.* for more detail).

Visual inspection supplemented quantitative data, providing insights into the policy's performance under varied conditions.

F. Measurements

Several measurements were used to evaluate and describe performance.

Step Count: Experiments were capped at 100 steps, each comprising four actions. The step count reflects how efficiently the robot is grasping the object. It also serves to evaluate whether the 100 step cap is reasonable, given the average number of steps needed to enact a successful grasp.

Gripper Close Count: The number of times the gripper closed and was subsequently re-opened because the succesful grasp criterium was not met. This was done to track the failures and provide more detailed data of how the robot is behaving during attempts. If the robotic arm is repeatedly closing and opening the gripper but not achieving a succesful grasp, it might be that its strategy for deciding when to grip is suboptimal.

Success Rate: Defined as both gripper fingers making contact with the object, regardless of grasp quality. This includes grasps where both fingers are making contact with the surface of the object, but without enabling the robotic arm to actually lift the object off the table's surface. This reflects the robot's ability to establish initial contact with the object, which is an important first step in grasping tasks. Since we expected the real-to-simulation gap to introduce some difficulty for the policy, we chose to, as a starting point, focus on this preliminary criterion as a means of evaluating the robot's initial ability to engage with objects.

Distance to Object: The smallest distance recorded between the arm and the object during an experiment. While absolute values were unreliable, relative comparisons provided useful insights. Initial positions placed the arm near the object, potentially limiting meaningful differences in smallest distance across objects.

III. Results

The percentage of succesful grasps per object can be seen in Figure 1.

For three of the objects, the policy attained a successful grasp for half or more than half of the trials. See Figure 2 for the average number of steps per trial for each of the objects. Here, we see that, when the arm knows how to grasp an object, the arm is relatively efficient in finding for the object. Importantly, it seems to be much faster than the cap of 100 timesteps. It seems to be the case, then, that the objects for which the average step count was high, had such a high step count because the gripper could not find the objects, rather than the gripper not having enough time to approach the objects.

In Figure 3, the amount of times the gripper met the close criterium and was then subsequently re-opened can be seen.

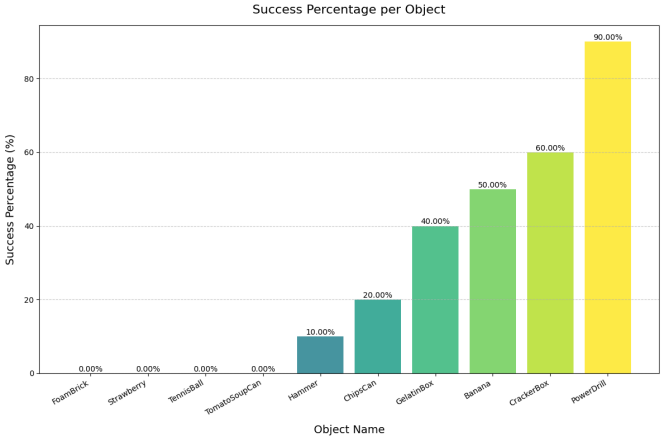

Fig. 1. Percentage of runs where the object was succesfully grasped.

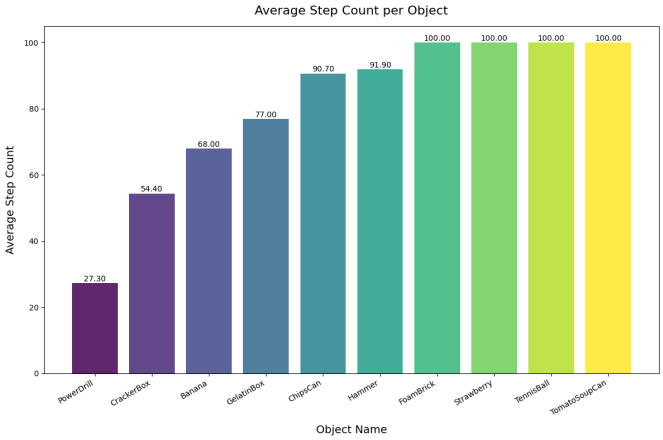

Fig. 2. Average amount of steps of 4 actions.

What is notable about this is that the amount of times the gripper was closed too soon was generally high for all objects. There seems to be a negative relationship between grasp success percentage and the average gripper close count, but the relationship does not seem to be very strong; there are some objects for which there was a very high gripper close count, but a non-zero success percentage, like the hammer.

Figure 4 shows the average smallest distance from the robotic arm to the object in question per trial for each object.

Results can be divided into three categories based on qualitative performance observations:

Category 1: Banana, Power Drill, Cracker Box: The policy consistently detected these objects and moved toward them with reasonable accuracy. Errors typically involved depth misjudgments or slight positional offsets. When objects were optimally spawned, performance was near perfect (See Figure 5).

Category 2: Strawberry, Tennis Ball, Tomato Soup Can: Performance was less reliable. For example, the policy often misidentified strawberries, initially moving correctly but then veering off course (see Figure 6).The language instructions tested (e.g., "pick up the red strawberry" vs. "pick up the

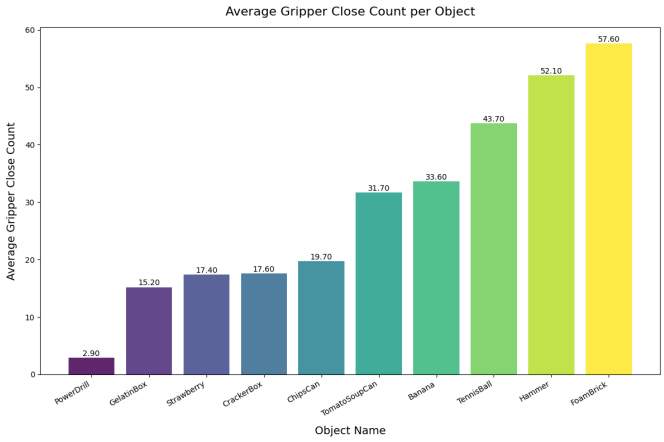

Fig. 3. Average amount of times the gripper was closed and then reopened.

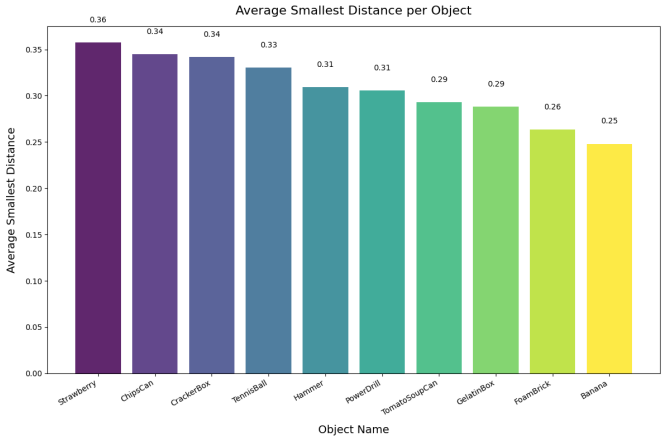

Fig. 4. Average smallest distance of the robot's gripper to the object throughout the run.

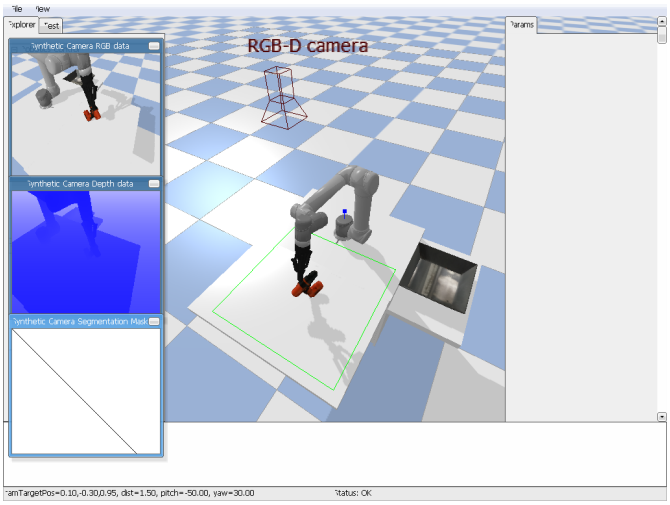

Fig. 5. Example grasp of the drill object. Note: the original position of the RGB-D camera illustrated above was moved to match the dataset that the CrossFormer was trained on more closely. The actual camera view can be seen in the top left.

red object", "pick up the strawberry", "pick up the object") showed no impact on grasp success.

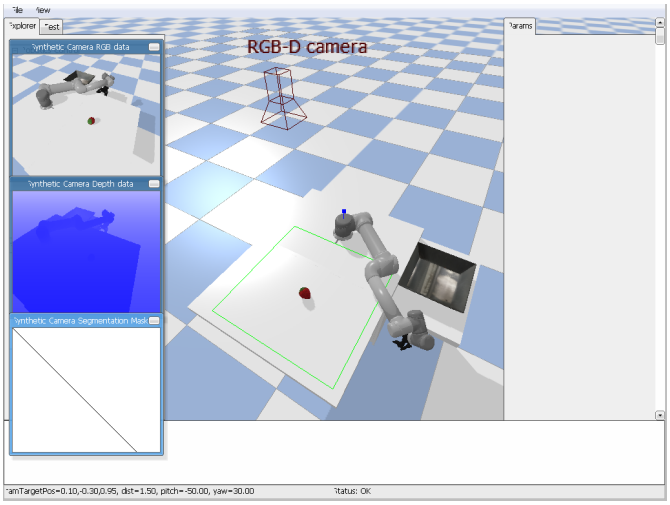

Fig. 6. Robotic arm veering off course, after having moved toward the strawberry.

Category 3: Foam Brick, Hammer: The policy consistently failed to grasp these objects, showing no initial signs of recognition (see Figure 7).

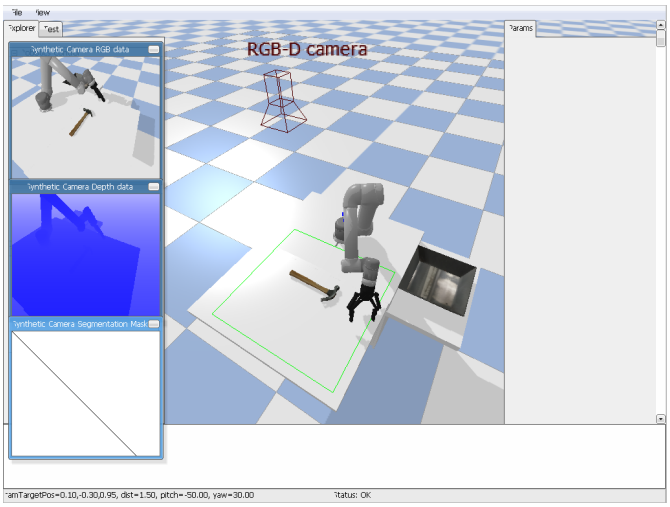

Fig. 7. Robotic arm showing no sign of recognizing and being able to approach the hammer object.

The highest success rate was achieved with the power drill, though some grasps were only technically successful. Visual inspection revealed that the policy performed best when objects were oriented vertically relative to the camera. The foam brick yielded the worst results, with erratic movements observed during virtually every run.

IV. Discussion

Though the results of this study demonstrate promising performance of a pretrained cross-embodiment learning architecture in a simulated robotic arm environment, domain gaps

and dataset biases significantly impair performance. These findings suggest that fine-tuning, even with minimal data, is needed to address key performance deficiencies.

A. Interpretation of Findings

First, the required alignment suggests a potential bias in the CrossFormer model, which is that the robot arm needs to be visible in the camera's field of view, and thus cannot rely solely on proprioceptive information.

A notable observation is the variability in gripper closure success across different objects. While the overall gripper closure count was high, success rates varied significantly depending on the object type. This may be a reflection of the challenges posed by domain gaps between the simulated environment and real-world data that the policy was trained on. For instance, the simulated strawberry might diverge significantly in appearance and physical characteristics from its real-world counterpart, potentially impairing the pretrained model's ability to plan its grasp.

The failure to grasp certain objects, such as the tennis ball (0% success rate), may suggest a sensitivity of the model to object geometry. While spherical objects like tennis balls may intuitively seem simple to grasp, the policy appeared to be optimized for elongated objects, such as the banana or the cracker box. However, our camera and arm configuration were calibrated around the banana and potentially its specific orientation toward the camera. The policy performed best when the object's longest side was vertically aligned with the camera's viewpoint. Given our experimental set-up, it is not possible to discern whether this is because of our calibration process or because of biases in the CrossFormer training. Regardless, these biases all may have hindered performance on rounded objects.

The model frequently seemed to misjudge the spatial relations and depth of the simulated environment. This depth perception likely explains the generally high number of attempted grasps per run. Even the object with a 90% grasp success rate and with the least amount of average steps had an average gripper close count of 2.90. During visual inspection, it could also be seen that the gripper was closing at moments where, from the camera's point of view, it would indeed make contact with the object, but not actually doing so, providing further evidence for the presence of a depth issue. This way, objects were consistently identified but not successfully grasped, suggesting that the model's grasp planning relied on imprecise depth estimations. This recurring issue may explain the inability to execute successful grasps despite object recognition. This is not to suggest a total absence of depth awareness, but rather a failure to fully adapt to the spatial nuances of the simulated domain, likely exacerbated by the absence of fine-tuning.

If depth perception were the only issue, however, we would not expect to see the robotic arm veering off course entirely, after initially moving towards the object. The fact that we observed this behavior may suggest that zero-shot generalization may take place in initially approaching the object, but that grasp planning relies too much on the specific embodiment and the object that is to be grasped. If the object and/or embodiment is not adequately represented in the cross-embodiment model, then the policy may not be able to execute grasp planning in that new embodiment without any fine-tuning.

B. Limitations

While this study provides interesting insights into the feasibility of zero-shot generalization of the CrossFormer architecture, it had significant limitations. First, we aligned the simulated environment with the dataset that the cross-embodiment policy was trained on, thus introducing a reduction in the extent to which the policy needed to generalize to this specific embodiment. However, the specific robot morphology and the fact that the environment is simulated likely still requires significant generalization on the part of the policy. Still, while we opted to calibrate the camera and arm position to improve performance, future studies could explore a more 'clean' zero-shot generalization approach that avoids this alignment altogether. Additionally, the calibration process was based solely on the Bridge v2 dataset, which accounts for only 17% of the OpenX Embodiment dataset [1]. This limited representation may have led the calibration process to a suboptimal outcome. Moreover, the best camera angle and robotic arm starting position were only based on runs using the banana as the object. Such idiosyncrasies further restrict the interpretability and transferability of the results. Moreover, our methodology for evaluating grasp success could benefit from more rigorous quantification. For example, the grasp success criterion may be somewhat misleading, given that it included grasps that would not allow the robot to lift the object. Finally, we lacked a systematic approach to assessing whether language instructions played a significant role in performance. While qualitative observations provided useful insights, future research could stand to benefit from a more systematic evaluation to draw definitive conclusions. In conclusion, this study was fundamentally exploratory in nature, and lacked a rigorous and systematic framework, which future research could adress.

C. Directions for Future Research

Since the policy already showed some ability to locate and grasp objects, future work could focus on fine-tuning the CrossFormer with just a single trajectory. As the policy seems to be sensitive to domain-specific nuances, this might already bring significant performance gains. This could then be compared against a zero-shot generalization baseline. Given the data that the CrossFormer was trained on, it would also be advisable to conduct these experiments on a physical robot. To ensure some alignment between the dataset and the camera of the new embodiment, a camera angle may be chosen by systematically varying the angle and then evaluating the robot's performance averaged over multiple objects placed in multiple locations. This could take the shape of a grid search. Trajectories of the robot over time may be tracked

systematically to derive insights its patterns of movement. The effect of different language instructions could also be systematically varied, not only for specific error cases, but for all cases. Future work could also evaluate different definitions of grasp success simultaneously, to obtain a more detailed understanding of any errors made by the policy. Further, understanding of the potential of zero-shot generalization of cross-embodiment policies may be deepened by conducting such experiments on different embodiments than the single-armed robot. This may be especially relevant, given the significant presence of the single-armed embodiment relative to other embodiments in the OXE dataset.

V. Conclusions

This study explored the zero-shot performance of a pre-trained cross-embodiment learning architecture deployed on a simulated robotic arm without fine-tuning. Our results suggest that performance is highly contingent on the idiosyncrasies of the training data and is significantly influenced by the gap between deployment and training. Still, our findings do not negate the promise of cross-embodiment learning, particularly its potential for generalizability and robustness; the fact that a cross-embodiment policy can be used on a robotic arm in a simulated environment to succesfully make contact with objects is certainly impressive. Our results suggest that fine-tuning, while it may not be strictly necessary, plays a critical role in bridging these domain gaps. Encouragingly, the required degree of fine-tuning may be minimal, which aligns with the goal of creating robotic learning systems that are scalable and adaptable.

Author's Contribution

The writing and coding tasks were evenly distributed, ensuring that each group member contributed sufficiently. We would also like to give a special thanks to both **Mohammadreza Kasaei** and **Georgios Tziafas** for their incredible support throughout the project.

References

[1] R. Doshi, H. Walke, O. Mees, S. Dasari, and S. Levine, "Scaling cross-embodied learning: One policy for manipulation, navigation, locomotion and aviation," arXiv preprint arXiv:2408.11812, 2024.

[2] J. Yang, C. Glossop, A. Bhorkar, D. Shah, Q. Vuong, C. Finn, D. Sadigh, and S. Levine, "Pushing the limits of cross-embodiment learning for manipulation and navigation," arXiv preprint arXiv:2402.19432, 2024.

[3] A. O'Neill, A. Rehman, A. Gupta, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, A. Tung, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, A. Gupta, A. Wang, A. Kolobov, A. Singh, A. Garg, A. Kembhavi, A. Xie, A. Brohan, A. Raffin, A. Sharma, A. Yavary, A. Jain, A. Balakrishna, A. Wahid, B. Burgess-Limerick, B. Kim, B. Schölkopf, B. Wulfe, B. Ichter, C. Lu, C. Xu, C. Le, C. Finn, C. Wang, C. Xu, C. Chi, C. Huang, C. Chan, C. Agia, C. Pan, C. Fu, C. Devin, D. Xu, D. Morton, D. Driess, D. Chen, D. Pathak, D. Shah, D. Büchler, D. Jayaraman, D. Kalashnikov, D. Sadigh, E. Johns, E. Foster, F. Liu, F. Ceola, F. Xia, F. Zhao, F. Vieira Frujeri, F. Stulp, G. Zhou, G. Sukhatme, G. Salhotra, G. Yan, G. Feng, G. Schiavi, G. Berseth, G. Kahn, G. Yang, G. Wang, H. Su, H. Fang, H. Shi, H. Bao, H. Ben Amor, H. Christensen, H. Furuta, H. Bharadhwaj, H. Walke, H. Fang, H. Ha, I. Mordatch, I. Radosavovic, I. Leal, J. Liang, J. Abou-Chakra, J. Kim, J. Drake, J. Peters, J. Schneider, J. Hsu, J. Vakil, J. Bohg, J. Bingham, J. Wu, J. Gao, J. Hu, J. Wu, J. Wu, J. Sun, J. Luo, J. Gu, J. Tan, J. Oh, J. Wu, J. Lu, J. Yang, J. Malik, J. Silvério, J. Hejna, J. Booher, J. Tompson, J. Yang, J. Salvador, J. Lim, J. Han, K. Wang, K. Rao, K. Pertsch, K. Hausman, K. Go, K. Gopalakrishnan, K. Goldberg, K. Byrne, K. Oslund, K. Kawaharazuka, K. Black, K. Lin, K. Zhang, K. Ehsani, K. Lekkala, K. Ellis, K. Rana, K. Srinivasan, K. Fang, K. Singh, K. Zeng, K. Hatch, K. Hsu, L. Itti, L. Chen, L. Pinto, L. Fei-Fei, L. Tan, L. Fan, L. Ott, L. Lee, L. Weihs, M. Chen, M. Lepert, M. Memmel, M. Tomizuka, M. Itkina, M. Guaman Castro, M. Spero, M. Du, M. Ahn, M. Yip, M. Zhang, M. Ding, M. Heo, M. Srirama, M. Sharma, M. Kim, N. Kanazawa, N. Hansen, N. Heess, N. Joshi, N. Suenderhauf, N. Liu, N. Di Palo, N. Shafiullah, O. Mees, O. Kroemer, O. Bastani, P. Sanketi, P. Fagan, P. Mitrano, P. Sermanet, P. Abbeel, P. Sundaresan, Q. Chen, Q. Vuong, R. Rafailov, R. Tian, R. Doshi, R. Martín-Martín, R. Baijal, R. Scalise, R. Hendrix, R. Lin, R. Qian, R. Zhang, R. Mendonca, R. Shah, R. Hoque, R. Julian, S. Bustamante, S. Kirmani, S. Levine, S. Lin, S. Moore, S. Bahl, S. Dass, S. Sonawani, S. Tulsiani, S. Song, S. Xu, S. Haldar, S. Karamcheti, S. Adebola, S. Guist, S. Nasiriany, S. Schaal, S. Welker, S. Tian, S. Ramamoorthy, S. Dasari, S. Belkhale, S. Park, S. Nair, S. Mirchandani, T. Osa, T. Gupta, T. Harada, T. Matsushima, T. Xiao, T. Kollar, T. Yu, T. Ding, T. Davchev, T. Zhao, T. Armstrong, T. Darrell, T. Chung, V. Jain, V. Kumar, V. Vanhoucke, W. Zhan, W. Zhou, W. Burgard, X. Chen, X. Chen, X. Wang, X. Zhu, X. Geng, X. Liu, X. Liangwei, X. Li, Y. Pang, Y. Lu, Y. Ma, Y. Kim, Y. Chebotar, Y. Zhou, Y. Zhu, Y. Wu, Y. Xu, Y. Wang, Y. Bisk, Y. Dou, Y. Cho, Y. Lee, Y. Cui, Y. Cao, Y. Wu, Y. Tang, Y. Zhu, Y. Zhang, Y. Jiang, Y. Li, Y. Li, Y. Iwasawa, Y. Matsuo, Z. Ma, Z. Xu, Z. Cui, Z. Zhang, and Z. Fu, "Open X-Embodiment: Robotic learning datasets and RT-X models: Open X-Embodiment Collaboration 0," in Proc. 2024 IEEE Int. Conf. Robot. Autom. (ICRA), 2024, pp. 6892–6903.