Robust Action Chunking Through Dynamic Action Selection

Hristo Todorov Matthew Li Vivien Cheng

CS224R: Deep Reinforcement Learning | Spring 2025 | Stanford University

Problem

Fine-manipulation tasks, such as transferring objects or threading cables are notoriously difficult for robots, because they require precise coordination of multiple components, and closed-loop visual feedback. Imitation learning methods have shown promise in such areas, as the sparse reward distribution prevents general reinforcement learning methods from being successful.

However, most imitation learning methods suffer from

- **Distribution shifts.** Small errors compound and lead to unexplored states and therefore the policy makes mistakes.
- **Non-stationary environments and demonstrations.** Environments are rapidly changing, and it is difficult to build general policies that adapt to unforeseen circumstances, or model inconsistent human behaviors.
- Additionally, **tackling one of those two problems amplifies the other**: one of the main ways to deal with distribution shifts is through **action chunks** or other **high-level action representations**, which makes the policy **less adaptive to rapid changes** in multiple small timesteps. In this project, we study how we can try **solving both problems simultaneously**.

Related Works

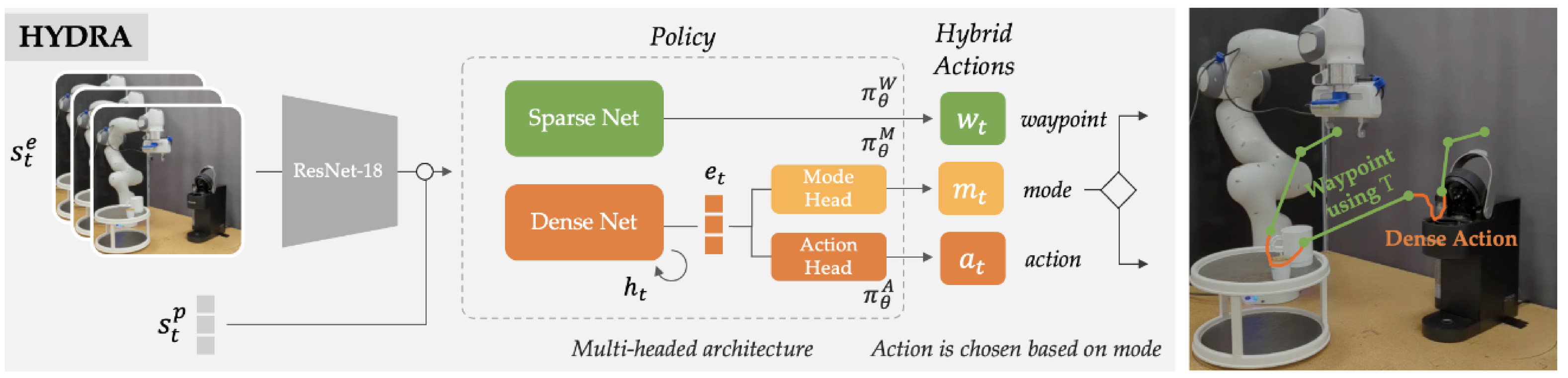

Figure 1. HYDRA tackles distribution shifts by decomposing actions into two categories: sparse high-level waypoints (e.g. get to the object) and dense low-level actions (e.g. grab the object). A different policy is trained on each category, and another policy is trained to switch between them.

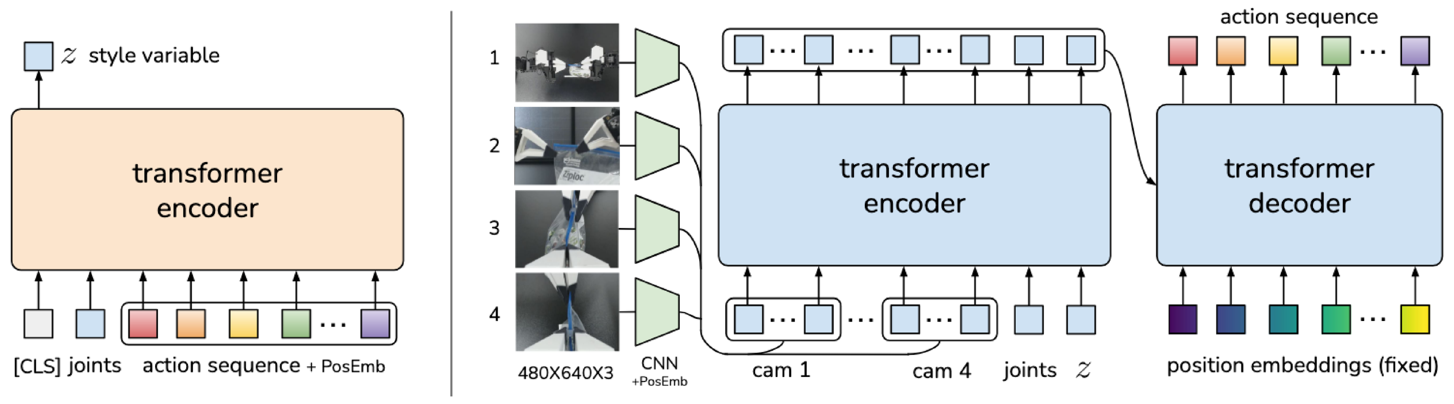

Figure 2. Action Chunking with Transformers (ACT) learns a generative model over action sequences, thus making a higher level action representation which reduces the effective planning horizon and thus reduces compounding errors.

Problems & Motivation

Even though both **HYDRA and ACT reduce distribution shifts** through higher-level action abstractions, both of them **fail when environments change rapidly**, as the preplanned waypoints (HYDRA) or action chunks (ACT) fail to capture the new changes.

In this project, we tackle ACT's main pitfall: robust action chunking. We propose a HYDRA-inspired action chunk selection that improves policy performance in rapidly changing environments..

Approach: Dynamic Action Chunking

HYDRA uses **high-level abstractions** for generally **easier** subtasks (e.g. waypoint movement to target) and **lower-level abstractions** for **harder** subtasks (e.g. fine-grained manipulation). However, HYDRA only defines **two action modes**, and trains a separate policy to switch between them. We **generalize HYDRA's idea to ACT**, where we use **model agreement to determine action chunk lengths**. We train two separate instances of ACT with different seeds, and look at the **normalized Euclidean distance between their action outputs** for a given observation to determine whether we should **truncate** an action chunk (models **disagree**) or **keep its large original size** (models **agree**).

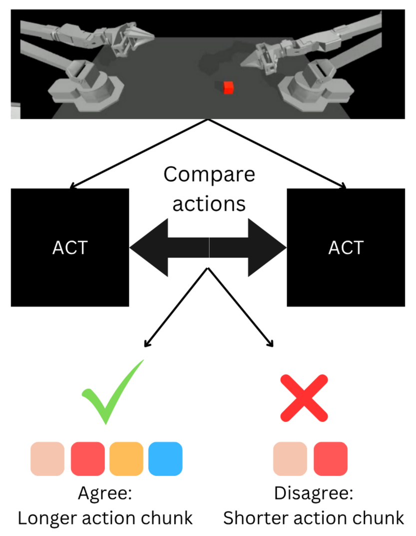

Figure 3. Dynamic action chunking: model agreement could determine subtask difficulty.

Approach: Outlier removal

To smooth actions, ACT introduces **temporal aggregation**, where a new action chunk is predicted at each time step and the action at any given timestep is an **exponentially weighted moving average** of all previous still relevant action chunks. However, we anticipate that environment shifts might lead to **stale action chunks**, inspiring outlier detection methods. At each timestep, we collect the distribution of all action chunks and calculate the sample **Mahalanobis distance**

$$d_M(\vec{v}) = \sqrt{(\vec{v}-\mu)^T \Sigma^{-1} (\vec{v}-\mu)} \quad (1)$$

of each action. Here, μ and Σ are the mean and covariance matrix of the distribution respectively. We then compute **IQR thresholds** based on these distances with a weighting towards the more recent actions and **remove outlier actions** from further consideration.

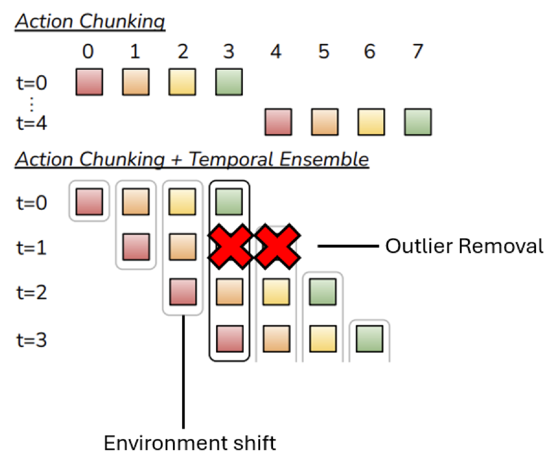

Figure 4. In temporal aggregation, environment shifts might lead to stale actions, leaving outliers which should be removed.

Future Work

We plan to use distribution width as a proxy for task difficulty in order to better select action chunk length. One limitation is that we weren't able to run evaluations in more stochastic environments like the real world, where we can interact with the robot. We expect both outlier removal and dynamic action chunking to show greater improvements from the baseline model in these scenarios.

Results

We trained and evaluated ACT on the transfer cube environment with human demonstrations from the ALOHA paper. The goal is to grab a cube with one robotic hand and transfer it in the air to another one.

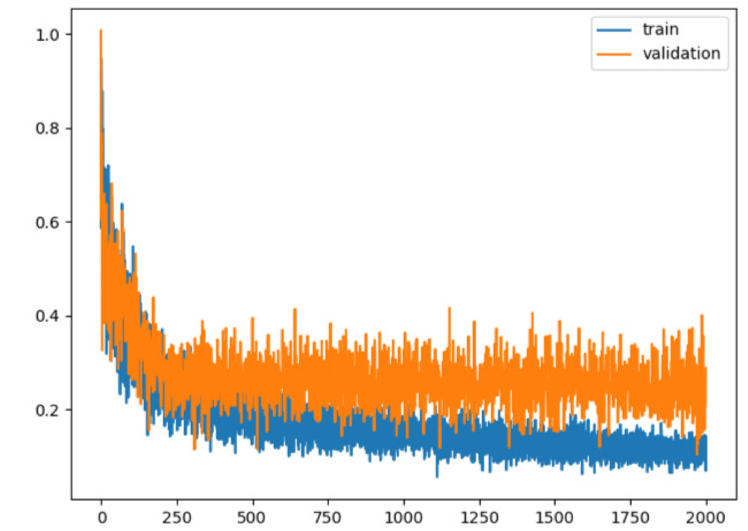

Figure 5. Training/Validation loss of ACT.

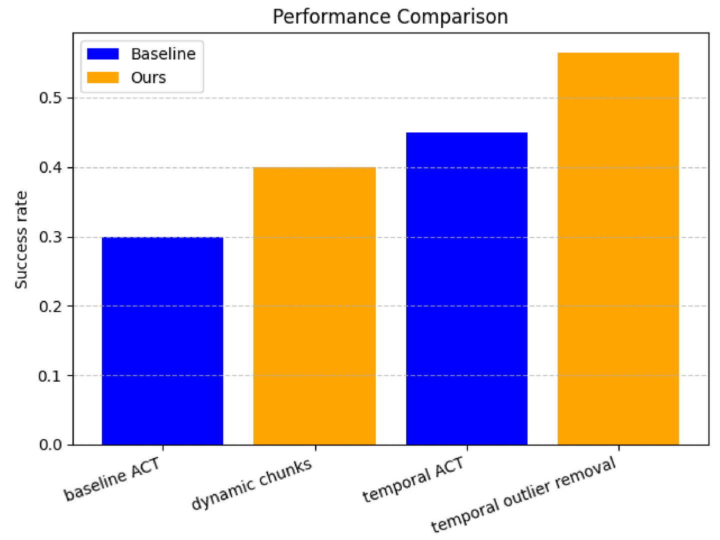

Figure 6. Dynamic chunks > ACT, Outlier Removal > Temporal ACT

Analysis

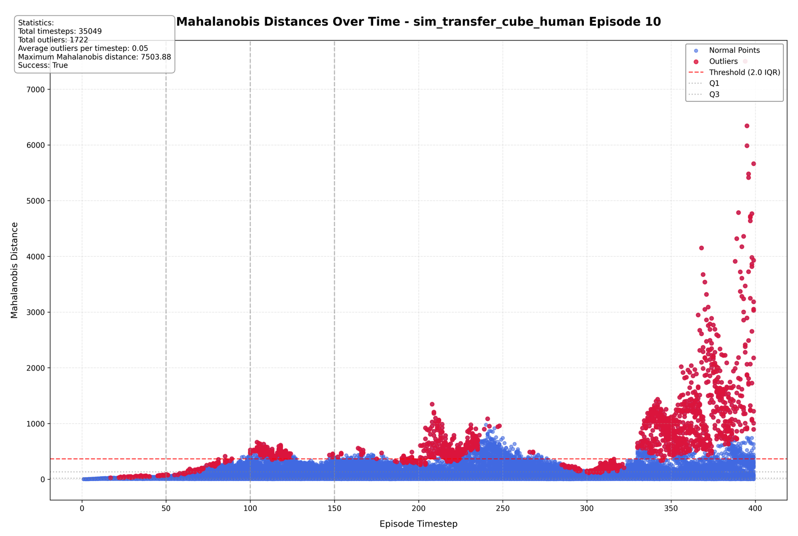

Figure 7. Success: Mahalanobis Distance

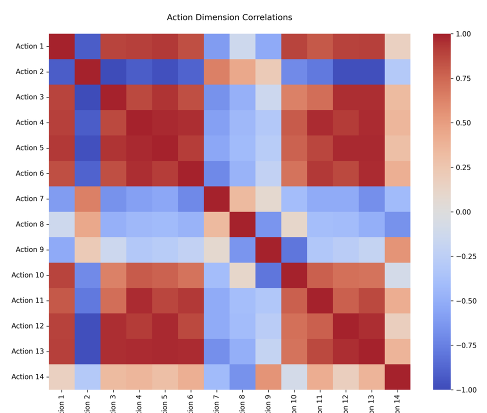

Figure 8. Success: Action Correlation Heatmap

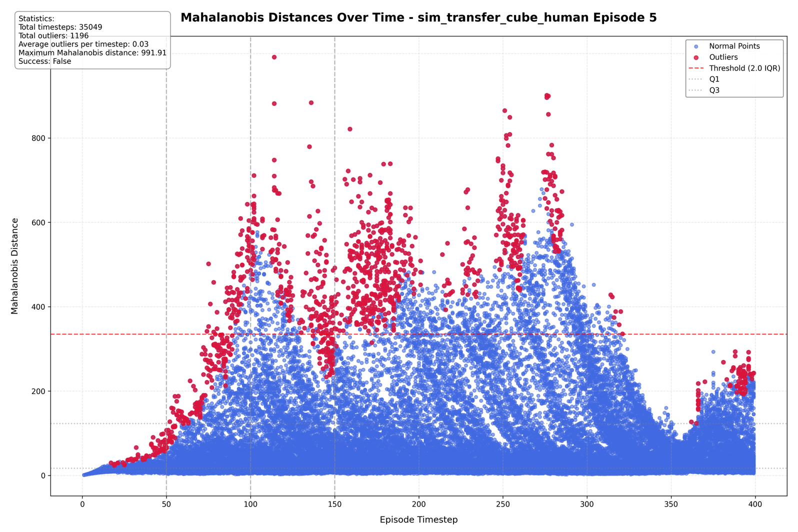

Figure 9. Failure: Mahalanobis Distance

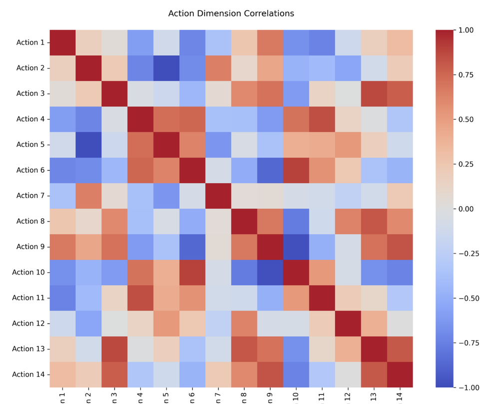

Figure 10. Failure: Action Correlation Heatmap

Successful episodes: tightly clustered Mahalanobis distances, indicating stable, in-distribution behavior and smooth, highly correlated action heatmaps, where the robot performs consistent joint movements.
Failed episodes: wider spread in Mahalanobis distances reflecting unstable predictions, and fragmented heatmaps with low correlation, suggesting the robot is reacting inconsistently across joints.

References

1. Zhao TZ, Kumar V, Levine S, Finn C. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware. arXiv. Published April 26, 2023. Accessed June 4, 2025. https://arxiv.org/abs/2304.13705.
2. Belkhale S, Cui Y, Sadigh D. HYDRA: Hybrid Robot Actions for Imitation Learning. arXiv. Published June 29, 2023. Accessed June 4, 2025. https://arxiv.org/abs/2306.17237