2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Viewpoint-Agnostic Manipulation Policies with Strategic *Vantage* Selection

Sreevishakh Vasudevan,[1] Som Sagar,[1] and Ransalu Senanayake[1]

***Abstract*— Since vision-based manipulation policies are typically trained from data gathered from a single viewpoint, their performance drops when the view changes during deployment. Naively aggregating demonstrations from numerous random views is not only costly but also known to destabilize learning, as excessive visual diversity acts as noise. We present *Vantage*, a viewpoint selection framework to fine-tune any pre-trained policy on a small, strategically chosen set of camera poses to induce viewpoint-agnostic behavior. Instead of relying on costly brute-force search over viewpoints, *Vantage* formulates camera placement as an information gain optimization problem in a continuous space. This approach balances exploration of novel poses with exploitation of promising ones, while also providing theoretical guarantees about convergence and robustness. Across manipulation tasks and policy families, *Vantage* consistently improves success under viewpoint shifts compared to fixed, grid, or random data selection strategies with only a handful of fine-tuning steps. Experiments conducted on simulated and real-world setups show that *Vantage* increases the task success rate by ≈25% for diffusion policies, and yields robust gains in dynamic-camera settings. GitHub: `https://github.com/sreevishakhv/Vantage_Public`**

I. INTRODUCTION

Modern robot manipulation policies trained with visual inputs have achieved levels of precision and adaptability that were once considered far-fetched. Yet, behind this success lies a subtle but critical limitation: most policies learn to act correctly only from the camera viewpoint they were trained on. Even slight changes in viewpoints can reduce a confident policy to a failure. This brittleness arises because camera viewpoint has a decisive influence on what a robot sees. For instance, an overhead view may offer an unobstructed global view of a workspace but risks being occluded as the arm moves. Similarly, a side-mounted camera tracks motion well but distorts object geometry, making attributes like orientation of objects harder to infer. More critically, in dynamic camera settings such as in humanoid robots with constantly moving heads, mobile manipulators with the camera mounted on pan-tilt heads or mobile bases, robots on moving assembly lines, viewpoint-specialized policies quickly lose their reliability, restricting robots from performing robustly outside carefully staged environments [1], [2].

One might expect that training on data from many random viewpoints would solve the problem. In practice, however, excessive variation from uninformative perspectives such as oblique, occluded, and overhead views act as corrupt data that hinders the model from learning stable, viewpoint-invariant representations. This phenomenon parallels the challenges of domain generalization, where exposure to overly diverse training distributions can hinder rather than help performance [3], [4]. Recent benchmarks also highlight how varying viewpoints affect models' ability to understand properties, affordances, and constraints in the workspace [5]. These challenges warrant new methods that strategically select informative viewpoints while avoiding those that hinder policy learning. Since the landscape of robot learning is shifting toward fine-tuning generalist policies [6] for specific applications, it becomes essential to perform this strategic selection during fine-tuning. Further, from a machine learning perspective, fine-tuning on a good enough model is more stable and data-efficient than learning to be invariant from

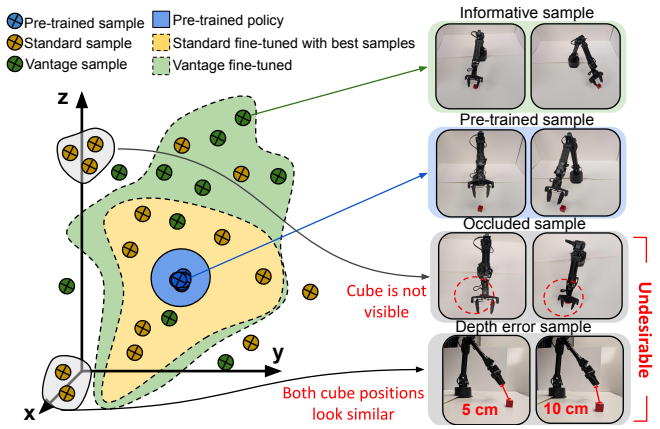

Fig. 1: Illustration of the viewpoint selection problem. Consider a setup where a camera can be placed anywhere in the $x - y - z$ space to observe the manipulator and its surrounding. Colored regions indicate where the camera can be placed for each policy for good performance. Pre-trained policies work only when a camera is placed in the narrow region (blue) of high accuracy, where it was originally trained. Standard fine-tuning (yellow), which relies on samples (i.e., collecting demonstrations) collected from randomly or uniformly placed camera viewpoints, spreads demonstration collection and fine-tuning budget across many *uninformative* regions. Such samples with occlusion or errors in depth perception (gray) can even hinder learning performance. *Vantage*, in contrast, strategically selects a small number of informative viewpoints (green), targeting areas that maximize downstream task performance. This allows vantage-fine-tuned policies to perform well even in dynamic camera settings, described in Section I.

[1]Laboratory for Learning Evaluation and Naturalization of Systems (LENS Lab), Arizona State University (ASU), USA. Emails: <svasud23,ssagar6,ransalu>@asu.edu

scratch. In this regime, the key question is no longer "How can we learn from every possible viewpoint?" but rather "Which viewpoints are most valuable for fine-tuning?"

We introduce Vantage, a framework that iteratively suggests camera poses for targeted data collection, enabling strategic fine-tuning that improves policy success rate across diverse camera poses. At its core, Vantage formulates viewpoint identification as a Bayesian optimization (BO) problem with a Gaussian process surrogate, optimizing a batched Upper Confidence Bound (q-UCB) acquisition strategy that suggests multiple informative viewpoints to collect data. Since collecting demonstrations from different views is time consuming, this approach provides a sample-efficient solution. By iteratively fine-tuning on these "vantage points," the policy learns to generalize across unseen perspectives, balancing view diversity with learning stability.

We evaluate Vantage across a real-world Unitree D1 arm and a simulated Panda arm on RoboSuite manipulation tasks using multiple vision-guided policy architectures including Behavior Cloning (BC), Batch-Constrained Deep Q-learning (BCQ), BC Transformers (BCT), Action Chunking Transformers (ACT), and Diffusion Policies. Our experiments reveal substantial and consistent gains: for instance, a Diffusion policy on Pick&Place improves from 37.01% to 83.20% success after fine-tuning with Vantage (+46.19%). These gains emerge with only a handful of fine-tuning steps, making the approach not merely effective but also sample-efficient. Our key contributions can be summarized as:

1) Proposing a general framework for learning viewpoint-agnostic manipulation policies.
2) Providing practically useful theoretical guarantees on the number of trials needed, stopping criteria for fine-tuning, and robustness under camera placement errors.
3) Extensive empirical validation to demonstrate that Vantage consistently outperforms baselines.

II. RELATED WORK

Early work sought robustness by training the policy with diverse variations in data such as changes in lighting, texture, background, and camera pose, so that the variations during training look similar to that of deployment (e.g., a shifted viewpoint) [7], [8]. Rather than physically capturing every camera pose, which is expensive, recent work attempts to synthesize alternative views through generative models. For instance, single-image novel-view diffusion models like Zero-1-to-3 [9], ZeroNVS [10] can render consistent target viewpoints. Robotics-specific pipelines such as VISTA [11] then use these renders to train policies that are more stable under camera shifts, or to inject corrective visual perturbations via scene reconstructions. Cross-embodiment viewpoint augmentation [12] has likewise been explored to transfer skills with fewer new samples. These methods have focused on extending the viewpoint coverage only by a small angle. Scaling these to different or larger observation spaces requires training a separate view-generating diffusion model for each small region, which limits their practicality due to extremely high computational cost. Such limitations warrant selecting only important viewpoints.

Another reason to be deliberate about viewpoint selection comes from evidence in large in-the-wild datasets such as DROID [13], which was originally designed to demonstrate the value of broad scene diversity, including varied setups that implicitly alter viewpoints. Nevertheless, recent analyses caution that overly aggressive augmentation or randomization can degrade accuracy and do not consistently improve generalization [3], [4], [14], motivating more targeted data sampling strategies. Parallel work in domain generalization further highlights that excessive diversity between training and test distributions can harm performance [4], [14]. Together, these findings underscore the importance of actively selecting informative viewpoints.

In 3D reconstruction, there are attempts to actively gather viewpoints during training [15], [16], [17], [18], [19], [20]. These methods focus on computing the similarity between two images as a proxy for information gain. However, this strategy does not transfer to manipulation settings, where distinct viewpoints do not necessarily translate into improved task performance. Rather than optimizing a single next view, our method based on BO emphasizes maximizing the downstream task performance of learned policies across unseen and dynamic camera settings. While BO has previously been used in robotics for informative path planning in mobile robots and controls [21], [22], to the best of our knowledge, it has not been used for viewpoint selection and finetuning.

III. METHODOLOGY

A. Viewpoint Selection as Optimization

At the heart of our framework lies a simple question: *from which camera angles should the robot learn in order to be robust across all others?* We define a viewpoint, θ, in the 3D space as the camera placement around the robot (see Fig. 2). Given a pre-trained policy π, our goal is to identify a **vantage point**, θ^{vantage}, from the set of all training points θ_{train}, such that fine-tuning π with data from this view maximizes the average task success rate $J\left(\pi_{\theta_{\text{train}}}; \Theta_{\text{test}}\right)$, across rollouts from a set of test viewpoints, Θ_{test}. Thus, we want to learn,

$$\theta^{\text{vantage}} = \underset{\theta_{\text{train}} \in \Theta_{\text{train}}}{\arg\max}\, J(\pi_{\theta_{\text{train}}}(\Theta_{\text{test}}))\,, \tag{1}$$

where, $\pi_{\theta_{\text{train}}}$ denotes the policy obtained by fine-tuning the pre-trained model on the original dataset together with demonstrations collected from viewpoint θ_{train}.

Since the mapping from viewpoint to downstream task success has no analytic gradients or closed-form structure, gradient-based optimization and exhaustive enumeration are impractical. This setting aligns precisely with the regime addressed by black-box optimization methods, motivating our choice of Bayesian optimization for efficient viewpoint selection in the continuous observation space of Θ_{test}. We will later establish in Theorem 3.1 that this formulation ensures the cumulative regret of our search remains sublinear, and hence, the number of poor viewpoint choices grows much slower than the number of fine-tuning attempts.

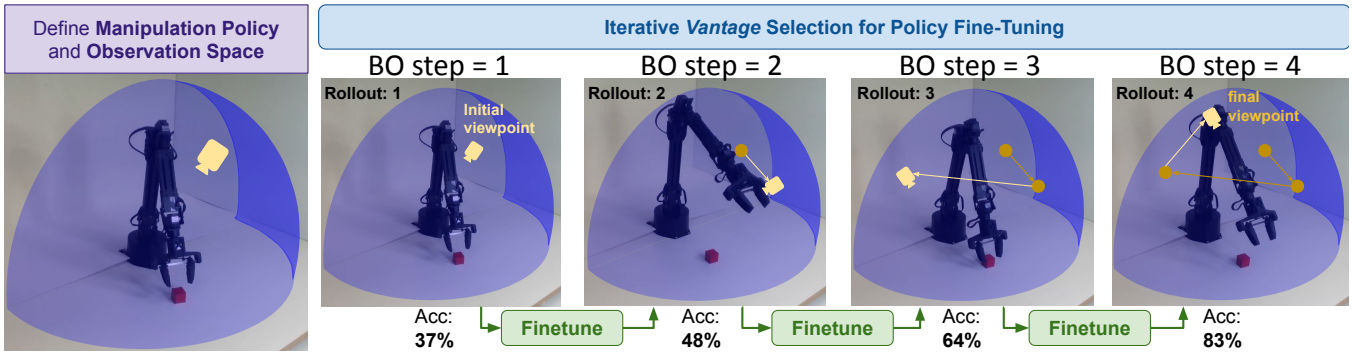

Fig. 2: Overview of the *Vantage* framework. Starting from a pre-trained manipulation policy and an initial camera viewpoint, the system iteratively selects additional viewpoints for fine-tuning. After each Bayesian optimization step (BO step), the updated policy is evaluated across the observation space, and the performance signal is used to guide the next selection. By progressively incorporating strategically chosen views, the policy becomes increasingly robust to viewpoint shifts, converging to a final model that generalizes across diverse views.

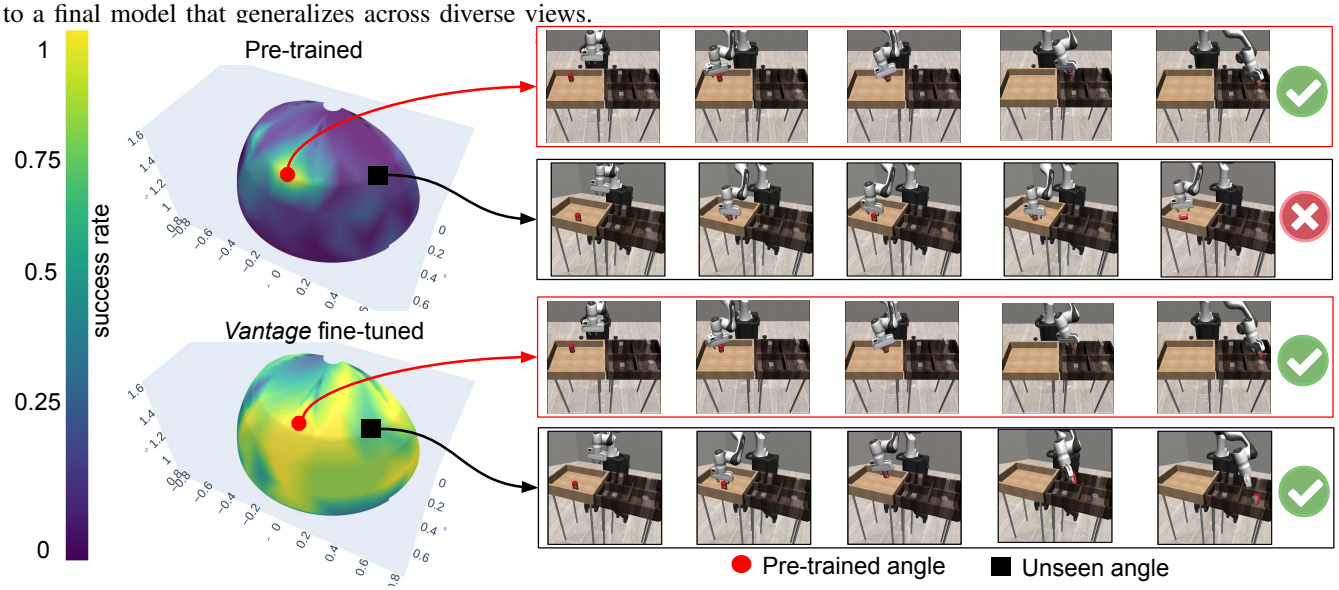

Fig. 3: Success rate across observation space (i.e., the cradle of a hemisphere where the camera can be moved at deployment). The top row shows performance under the default pretraining viewpoint versus a novel test viewpoint. The bottom row compares the same hemisphere after *Vantage* fine-tuning. While the pre-trained policy works around only where it was trained, *Vantage*-fine-tuned model works almost everywhere.

B. Bayesian Optimization for Viewpoint Search

We model the mapping from training viewpoint θ_{train} to performance J using a Gaussian Process (GP):

$$J(\pi_{\theta_{\text{train}}}) \sim \mathcal{GP}\big(\mu(\theta_{\text{train}}), k(\theta_{\text{train}}, \theta'_{\text{train}})\big), \tag{2}$$

with squared-exponential kernel $k(\cdot,\cdot)$. This surrogate captures our uncertainty: the posterior mean μ predicts expected manipulation task performance, while the covariance induced by k encodes uncertainty in predicting task performance due to similarity between camera positions. This uncertainty allows us to decide *where to look next* without exhaustively training on every view.

Classical BO would select one new viewpoint at a time, but policy fine-tuning can be parallelized, thus we use a batched acquisition strategy, q-Upper Confidence Bound (q-UCB) [23]. For a batch $\Theta_q = \{\theta_1, \ldots, \theta_q\}$, q-UCB scores each candidate set of viewpoints by combining exploitation (high predicted mean) with exploration (high uncertainty):

$$\begin{aligned}\Theta_q^{\text{next}} &= \underset{\Theta_q \subset \Theta_{\text{train}}}{\arg\max}\, \alpha_{\text{qUCB}}(\Theta_q) \\ &= \underset{\Theta_q \subset \Theta_{\text{train}}}{\arg\max}\, \mathbb{E}_{\tilde{\Theta} \sim \mathcal{N}\left(\mu(\Theta_q),\, \frac{\beta\pi}{2}\Sigma(\Theta_q)\right)} \\ &\quad \Big[\max_{i=1,\ldots,q} \big(\mu(\theta_i) + |\tilde{\Theta}_i - \mu(\theta_i)|\big)\Big].\end{aligned} \tag{3}$$

Here, $\mathcal{N}$ is a normal distribution and $\beta > 0$ controls exploration vs exploitation: large β encourage sampling from uncertain regions, while smaller ones favor viewpoints already predicted to perform well. Please refer Appendix A of [23] for the detailed derivation of (3). As we will show in Theorem 3.2, this acquisition strategy ensures that the average task success of policies fine-tuned with *Vantage* converges toward the optimal viewpoint, with error diminishing at rate $O(T^{-1/2})$ with T steps.

C. Iterative Fine-Tuning Procedure

The full Vantage procedure unfolds as follows:

1) **Initialization.** Sample q random viewpoints, fine-tune, and record success rates. Fit an initial GP surrogate.
2) **Vantage training loop.** In iteration T, update the GP with accumulated data, select a new batch of viewpoints via q-UCB, fine-tune at each, evaluate, and expand the dataset.
3) **Final selection.** After N rounds, choose the fine-tuned policy with the highest observed success rate.

This iterative process is given in detail in Algorithm 1. Conceptually, each round of fine-tuning both improves the policy and refines our estimate of the underlying function $f(\theta)$, steering the search toward increasingly informative vantage points. Moreover, practical deployment often faces camera placement noise due to calibration errors or mechanical offsets; Theorem 3.3 will show that the BO trajectory is provably robust to such Gaussian perturbations, ensuring stability of our method in real-world settings.

D. Theoretical Guarantees

Beyond empirical performance, Vantage provides provable guarantees that offer practical guidance for engineers. These results not only establish formal learning-theoretic bounds, but also translate into actionable principles for system design: how many trials are needed, when to stop fine-tuning, and how precise the hardware setup must be. By framing the guarantees in terms of data efficiency, convergence behavior, and robustness to camera placement errors, Vantage bridges the gap between abstract theory and the concrete engineering decisions required for real-world robot deployment.

Theorem 3.1 (Efficiency of viewpoint selection): Let $f : \Theta \to [0,1]$ denote the mapping from training viewpoints to average success rates, drawn from a GP prior with kernel k. Assume observations $y_t = f(\theta_t) + \varepsilon_t$ with ε_t sub-Gaussian. Running q-UCB for T rounds yields cumulative regret

$$R(T) = \sum_{t=1}^{T}\sum_{j=1}^{q}\big[f(\theta^*) - f(\theta_{t,j})\big] = O\Big(\sqrt{qT\,\gamma_{qT}\,\beta_T}\Big),$$

with probability at least $1-\delta$, where δ is a user chosen failure probability, and $\theta^* = \arg\max_{\theta\in\Theta} f(\theta)$, γ_{qT} is the batch information gain, and β_T is confidence.

Proof: [Proof sketch] The argument follows Theorem 2 of [24]. Each UCB choice ensures that the instantaneous regret is bounded by a multiple of the posterior standard deviation. Summing over qT queries and applying Cauchy–Schwarz with the information gain bound yields the sublinear regret rate. Full proof Appendix 1.1. ∎

This GP-UCB regret bound assures engineers that wasted trials grow slowly compared to total trials, meaning only a handful of optimized camera placements (e.g., 10–15) are typically sufficient rather than hundreds. This lets practitioners budget resources such as robot hours and data collection costs up front.

Algorithm 1 *Vantage*

Step 1: Gather Initial Data
Sample q random viewpoints $\{\theta_{\text{train}}^{(j)}\}_{j=1}^{q}$, where each $\theta_{\text{train}}^{(j)} \in \Theta_{\text{train}}$
Generate manipulation datasets $\{\mathcal{D}^{(j)}\}_{j=1}^{q}$, from robot trials or simulation at viewpoints $\{\theta_{\text{train}}^{(j)}\}_{j=1}^{q}$
Fine-tune the original policy on $\mathcal{D}_j$ independently
Evaluate the fine-tuned models across Θ to obtain success rates $\{J_i\}_{i=1}^{q}$
Initialize a historical dataset $\mathcal{D}_{gp} \leftarrow \{(\theta_j, J_i)\}_{i=1}^{q}$
Step 2: Vantage training loop
for $i = 1$ to N **do**
  Use D_{gp} and Bayesian optimization to select q new angles $\{\theta_{\text{new},j}\}_{j=1}^{q} = (\theta_h^{\text{new},j}, \theta_v^{\text{new},j}) \in \Theta$
  Generate datasets $\mathcal{D}_{\text{new},j}$ at $\theta_{\text{new},j}$
  Fine-tune the original model separately on $\mathcal{D}_{\text{new},j}$
  Evaluate the model fine-tuned at θ_{new} to obtain J_{new}
  Update $\mathcal{D}_{gp} \leftarrow \mathcal{D}_{gp} \cup \{(\theta_{\text{new},j}, J_{\text{new},j})\}$
end for
Step 3: Final Selection
$\theta^{vantage} \leftarrow \arg\max_{\theta\in\Theta} J(\pi_\theta)$

Theorem 3.2 (Success rate convergence): Under the same assumptions, with probability at least $1-\delta$,

$$\frac{1}{T}\sum_{t=1}^{T} J(\pi_{\theta_t}) \;\geq\; J(\pi_{\theta^*}) - O\Big(\sqrt{\tfrac{\gamma_T\,\beta_T}{T}}\Big).$$

Proof: [Proof sketch] From Theorem 3.1, cumulative regret is $O(\sqrt{T\,\gamma_T\,\beta_T})$. Dividing by T gives the average success rate bound, which shows that the mean performance of policies fine-tuned with Vantage converges to the optimum. Full proof is provided in the Appendix 1.2. ∎

These results guarantee that, as the number of fine-tuning rounds increases, Vantage converges to near-optimal camera viewpoints while requiring only sublinear exploration of the vast space. Engineers can use this to define stopping rules to avoid over-training and save time. Once success rates plateau, further fine-tuning offers diminishing returns.

Theorem 3.3 (Robustness under camera placement error): Let $f : \mathbb{R}^d \to \mathbb{R}$ be modeled with a Gaussian process prior using the squared–exponential covariance. Suppose Bayesian optimization selects query viewpoints $x_t \in \mathbb{R}^d$, but the executed viewpoints are $\tilde{x}_t = x_t + \varepsilon_t$, where $\varepsilon_t \sim \mathcal{N}(0, \sigma_x^2 I_d)$ are i.i.d. perturbations due to camera placement error of standard deviation σ. Define the effective objective $g(x) = \mathbb{E}[\,f(x+\varepsilon)\,]$. Then g is again governed by a squared exponential GP with re-parametrized hyperparameters. Hence, *Bayesian optimization with squared–exponential priors is robust to such input noise*: the optimization trajectory is unchanged relative to the noise-free case.

Proof: [Proof sketch] Since $g(x) = \mathbb{E}_\varepsilon[f(x+\varepsilon)]$ is a linear functional of a GP, g is again a GP. Its covariance is

$$k_g(x,x') = \mathbb{E}_{\varepsilon,\varepsilon'}\big[k(x+\varepsilon,\, x'+\varepsilon')\big],$$

TABLE I: Success rates (%) of policies across tasks under three camera settings: *Default* (trained viewpoint), Θ (set of all allowed viewpoints: quarter-sphere in front of the robot), and *Dynamic* (continuous motion). Results compare baseline training, grid/random augmentation, and our proposed method *Vantage*.

	Camera Placement	BC				Diffusion				BCT			
		Base	Grid	Standard	Vantage	Base	Grid	Standard	Vantage	Base	Grid	Standard	Vantage
Lift	Default	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
	Θ	6.90	9.18	23.15	**23.50**	50.40	58.57	65.82	**66.10**	6.71	10.12	10.12	**10.44**
	Dynamic	91.66	93.33	**100.0**	**100.0**	98.33	**100.0**	**100.0**	**100.0**	91.66	97.33	97.33	**98.12**
Pick Place	Default	70.0	70.0	70.0	70.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0
	Θ	0.80	1.04	2.53	**3.90**	37.01	22.09	66.11	**83.20**	1.69	1.70	1.94	**1.94**
	Dynamic	3.00	4.23	7.66	**9.88**	88.33	71.54	92.33	**97.16**	13.33	16.17	18.04	**18.04**
Square	Default	30.0	20.0	**30.0**	20.0	60.0	60.0	60.0	**70.0**	50.0	50.0	60.0	**60.0**
	Θ	0.24	0.61	0.74	**1.00**	2.38	2.38	12.08	**14.80**	0.74	1.49	1.49	**1.49**
	Dynamic	0.00	0.00	0.00	0.00	8.33	8.33	40.0	**54.66**	0.00	0.51	0.51	**0.51**

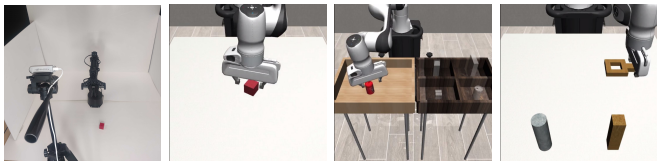

Fig. 4: Experiment setups for *Vantage*: (First) external camera placement with Unitree D1 arm, (second, third, fourth) RoboSuite environments for benchmark tasks.

with k the squared–exponential kernel. Because $\varepsilon - \varepsilon' \sim \mathcal{N}(0, 2\sigma_x^2 I_d)$, this expectation reduces to a Gaussian integral

$$k_g(x, x') = \sigma_f^2 \, \mathbb{E}_\delta\Big[\exp\big(-\tfrac{\|(x-x')+\delta\|^2}{2\ell^2}\big)\Big], \quad \delta \sim \mathcal{N}(0, 2\sigma_x^2 I_d).$$

Applying the standard Gaussian moment identity yields

$$k_g(x, x') = \sigma_f^2 \Big(\tfrac{\ell^2}{\ell^2 + 2\sigma_x^2}\Big)^{d/2} \exp\Big(-\tfrac{\|x-x'\|^2}{2(\ell^2+2\sigma_x^2)}\Big).$$

Thus g has the same squared–exponential form with enlarged lengthscale and reduced marginal variance as stated. Full proof is provided in the Appendix 1.3. ■

This theorem highlights that engineers do not need ultra-precise camera placement; even with modest calibration errors or approximate positioning, performance remains stable, highlighting the method's practical simplicity.

IV. EXPERIMENTAL RESULTS

We design experiments to rigorously evaluate *Vantage* across both simulation and real-robot settings. Our goal is to assess whether strategic viewpoint selection improves generalization to novel viewpoints and dynamic camera motion. In particular, we seek to answer the following four questions:

1) How does a policy fine-tuned with *Vantage* compare to standard and grid based fine-tuning? (Sec IV-B)
2) Does *Vantage* enable more efficient discovery of informative camera viewpoints? (Sec IV-C)
3) How well do *Vantage* trained policies work in dynamic viewpoint changes? (Sec IV-D)
4) Do the improvements achieved with *Vantage* extend to real robot deployments? (Sec IV-E)

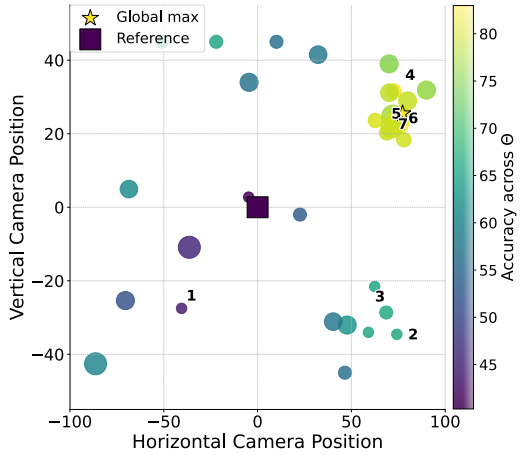

Fig. 5: Vantage iterations for viewpoint selection on *Pick & Place*. Each point corresponds to a candidate camera position evaluated during fine-tuning, with color and size indicating success rate and iteration, respectively. The purple square marks the default training viewpoint, while the yellow star denotes the globally best-performing *vantage point*. The search progressively concentrates around informative regions, converging to the optimum by iteration 7.

A. Setup

Simulation: We conduct experiments in the RoboSuite framework [25] using RoboMimic benchmark datasets [26], which provide demonstrations for three standard manipulation tasks: *Lift*, *Square*, and *Pick & Place*. To test generality across model classes, we benchmark *Vantage* with policies spanning diverse architectures and inductive biases, including BC [27], BCQ [28], BCT [26], and Diffusion [29]. Each policy is pre-trained on demonstrations collected from the default viewpoint in RoboSuite. We define the quarter-sphere in front of the robot (see Fig 2) as the space of allowed viewpoints (Θ). Then the policy is fine-tuned from different angles selected by various viewpoint selection strategies (default fine-tuning, grid search and *Vantage*).

Real robot: To study sim-to-real transfer, we deploy *Vantage* on a Unitree D1 6-DoF arm (see Fig 4). We fine-tune an ACT based [30] visuomotor policy on a reaching task while systematically varying camera placements around

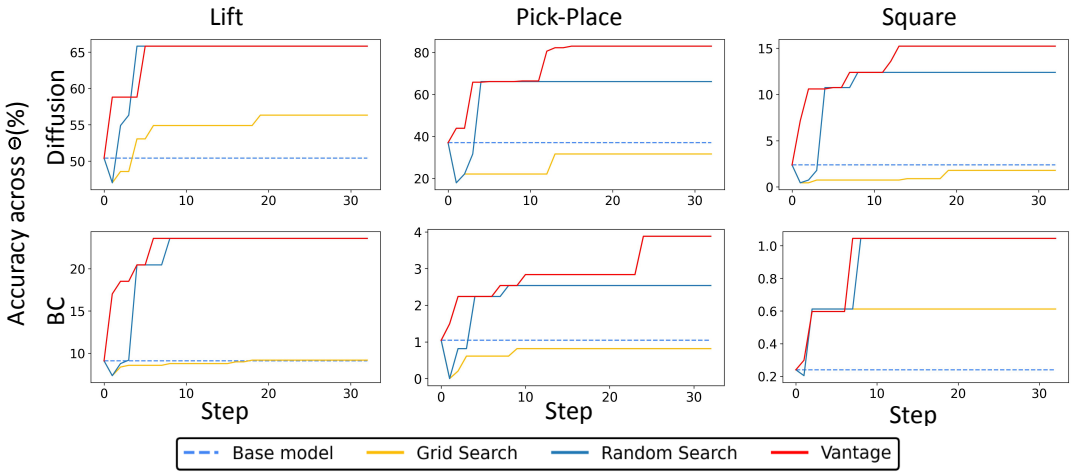

Fig. 6: Convergence of best accuracy across viewpoint space Θ for Diffusion policies on *Lift*, *Pick & Place*, and *Square*. Each curve tracks the best-performing policy over optimization steps under different search strategies: base model (dashed), grid search, random search, and *Vantage*. *Vantage* (red) consistently converges faster and achieves higher final accuracy with fewer iterations, demonstrating its sample efficiency compared to exhaustive or heuristic approaches.

the workspace. This setup allows us to test whether the robustness observed in simulation also persists under real-world sensing noise and actuation dynamics.

We evaluate under two regimes that reflect realistic deployment scenarios:

1) **Diverse static viewpoints:** Policies are tested across a quarter-sphere of feasible camera placements, probing generalization to novel but fixed viewpoints not seen during fine-tuning.
2) **Dynamic viewpoints:** Policies are tested under continuous camera motion during inference, simulating deployment with mobile manipulators or head-mounted cameras where viewpoints shift over time.

B. Comparison with Viewpoint Search Baselines

To contextualize performance, we compare against three standard baselines.

1) **Default angle accuracy** evaluates policies exclusively on the original pretraining viewpoint.
2) **Grid search fine-tuning** systematically samples viewpoints over a discretized grid, offering broader coverage but at significant cost.
3) **Standard viewpoint fine-tuning** samples views randomly, providing diversity but often wasting training budget on uninformative perspectives.

As summarized in Table I, *Vantage* consistently matches or outperforms these baselines across all tasks and policy families. The most dramatic gains appear in *Pick & Place* with a diffusion policy, where success improves from 37.01% to 83.20% (+46.19%). Similar trends are observed for BC and BCT, demonstrating robustness across architectures.

C. Efficiency of Vantage for Viewpoint Discovery

Unlike grid and random strategies, which waste budget on redundant or uninformative views, *Vantage* leverages Bayesian optimization to quickly identify promising viewpoints. As visualized in Fig. 5, the search concentrates in informative regions after only a few iterations. In practice, convergence occurs within 10–15 trials, a fraction of the budget required by exhaustive search. This efficiency is further reflected by the rapid performance gains as shown in Fig. 6. The q-UCB calculation to determine the next viewpoint requires less than 30 seconds, while the combined process of data collection and fine-tuning completes within $\approx$1 hour, underscoring the overall time efficiency of the approach.

TABLE II: Comparison of success rates (%) across augmentation strategies. Viewpoint Randomization with 1 or 8 additional views provides limited or inconsistent gains, sometimes degrades performance. Generated Augmentations with 5 synthetic views offers moderate improvements but remains less effective than optimized selection. In contrast, *Vantage* achieves competitive or superior performance with only a single optimized viewpoint.

Task & Policy	Viewpoint Randomization (1 view)	Viewpoint Randomization (8 views)	Generated Augmentation (5 views)	Vantage (1 view)
Pick Place (BC)	2.1	4.3	1.2	**3.9**
Pick Place (Diffusion)	63.8	42.6	52.3	**83.2**
Square (BC)	0.9	0.3	0.3	**1.0**
Square (Diffusion)	12.3	**17.5**	14.6	14.8
Real. (ACT)	36.0	30.0	—	**44.0**

Quantitatively, Table II further highlights that *Vantage* outperforms heavy augmentation. For example, in *Pick Place* with a diffusion policy, viewpoint randomization with 8 additional views degrades performance, while *Vantage* reaches 83.2% using only a handful of views. This efficiency is also evident in the optimization landscape of Fig. 3, which shows how *Vantage* reshapes the performance contours.

D. Performance Under Dynamic Viewpoints

A central test of real-world utility is robustness under dynamic camera motion. *Vantage* fine-tuned policies sustain

performance even when camera viewpoints change continuously at inference time. Dynamic viewpoint evaluation simulates settings such as mobile manipulators or head-mounted cameras, where the robot cannot rely on a fixed perspective. As shown in Table I, policies fine-tuned with *Vantage* consistently exhibit higher robustness under dynamic evaluation compared to baselines. For example, on *Pick & Place* with a Diffusion policy, success under dynamic motion increases from 88.3% to 97.2% with *Vantage*. On the more challenging *Square* task, Diffusion policy with *Vantage* improves in accuracy from 8.3% to 54.7%, demonstrating resilience to viewpoint ambiguity. Even simple BC policies, which otherwise collapse under motion (e.g., $3.0\% \rightarrow 9.9\%$ on *Pick & Place*), benefit from targeted viewpoint selection.

Table II further highlights this point by comparing *Vantage* against viewpoint randomization (VR) [7] and generated augmentation strategies akin to VISTA[11]. While VR with many views can sometimes improve performance (e.g., Diffusion on *Square*, $12.3\% \rightarrow 17.5\%$), it often requires large data budgets and may even degrade performance due to unstructured variability (VR with 8 views on *Pick & Place* reduces accuracy from 63.8% to 42.6%). In contrast, *Vantage* achieves 83.2% success with only a handful of optimized views, demonstrating sample efficiency and superior robustness. By explicitly optimizing where the robot learns to see from, *Vantage* induces viewpoint invariant representations that remain stable under continuous camera motion.

E. Does Vantage work on real-world robots?

Finally, we evaluate whether the gains achieved in simulation translate to physical hardware. We deploy *Vantage* on a Unitree D1 6-DoF arm and fine-tune an ACT based visuomotor policy for a reaching task under varying camera placements. As shown in Table II, naive augmentation strategies such as VR provide only limited or inconsistent improvements, with success rates ranging from 30.0% to 36.0%. In contrast, *Vantage* achieves a success rate of 44.0% using only a single optimized viewpoint, outperforming the other methods. Because camera placement is imprecise in practice, in simulation we also inject realistic placement noise and measure the post-*Vantage* accuracy of the best-performing model. This experiment revealed that, even under such placement errors, models trained with *Vantage* remained within $\approx$5% margin of error. These results confirm that strategic viewpoint selection not only improves robustness in simulation but also extends to real-world sensing and actuation conditions, where camera calibration errors, occlusions, and lighting variability pose additional challenges.

V. CONCLUSIONS

We introduced *Vantage*, a framework that treats camera viewpoint selection as a continuous optimization problem and leverages Bayesian optimization to identify informative perspectives for fine-tuning visuomotor policies. By explicitly balancing exploration of novel viewpoints with exploitation of high-performing ones, *Vantage* provides a principled and sample-efficient alternative to naive data augmentation or exhaustive viewpoint search. Our experiments across search benchmarks and multiple policy architectures demonstrate that *Vantage* consistently improves robustness to viewpoint shifts, often by large margins. In particular, we observe substantial gains even with limited fine-tuning, as well as strong generalization under dynamic camera evaluation. These results highlight that carefully chosen training viewpoints can be as important as the underlying policy architecture in achieving viewpoint-agnostic manipulation.

APPENDIX

A. Proofs of Theoretical Guarantees

We formalize the theoretical guarantees for *Vantage* in this section.

Theorem 1.1 (GP-UCB Regret Bound): Let $f : \Theta \rightarrow \mathbb{R}$ be a mapping from angles to success rates, drawn from a GP prior with kernel k. At each round $t = 1, \ldots, T$ choose,

$$\theta_t \;=\; \arg\max_{\theta \in \Theta} \Big[\mu_{t-1}(\theta) \;+\; \sqrt{\beta_t}\, \sigma_{t-1}(\theta) \Big],$$

and observe $y_t = f(\theta_t) + \varepsilon_t$ with ε_t zero-mean sub-Gaussian noise. Then, with probability at least $1 - \delta$,

$$R(T) \;=\; \sum_{t=1}^{T} \sum_{i=1}^{q} \big[f(\theta^*) - f(\theta_{t,j}) \big] \;=\; O\big(\sqrt{qT\, \gamma_{qT}\, \beta_T} \big),$$

where $\theta^* = \arg\max_{\Theta} f$, γ_T is the maximum information gain after T steps, and β_T is chosen as in [24].

Proof: By Theorem 2 of Srinivas *et al.* (2012), with probability $1 - \delta$, for all t and all θ,

$$\big| f(\theta) - \mu_{t-1}(\theta) \big| \;\le\; \sqrt{\beta_t}\, \sigma_{t-1}(\theta).$$

Hence the instantaneous regret $r_t = f(\theta^*) - f(\theta_t)$ satisfies

$$r_t \;\le\; 2\, \sqrt{\beta_t}\, \sigma_{t-1}(\theta_t).$$

Summing and applying Cauchy–Schwarz with with the definition of information gain gives

$$\begin{aligned} R(T) &= \sum_{t=1}^{T} r_t \le 2 \sum_{t=1}^{T} \sqrt{\beta_t}\, \sigma_{t-1}(\theta_t) \\ &\le 2 \sqrt{ \Big(\sum_t \beta_t \Big) \Big(\sum_t \sigma^2_{t-1}(\theta_t) \Big) } = O\big(\sqrt{T\, \gamma_T\, \beta_T} \big) \end{aligned}$$

■

Theorem 1.2 (Average Success Convergence): Under the same setting as Theorem 1.1, with probability at least $1 - \delta$,

$$\frac{1}{T} \sum_{t=1}^{T} J\big(\pi_{\theta_t} \big) \;\ge\; J\big(\pi_{\theta^*} \big) \;-\; O\Big(\sqrt{\tfrac{\gamma_T\, \beta_T}{T}} \Big).$$

In particular, the mean success converges to the optimum at rate $O(T^{-1/2})$.

Proof: From Theorem 1.1, with high probability,

$$\sum_{t=1}^{T} \big[J(\pi_{\theta^*}) - J(\pi_{\theta_t}) \big] = O\big(\sqrt{T\, \gamma_T\, \beta_T} \big)$$

$$\implies\; J(\pi_{\theta^*}) - \frac{1}{T} \sum_{t=1}^{T} J(\pi_{\theta_t}) = O\Big(\sqrt{\tfrac{\gamma_T\, \beta_T}{T}} \Big), f$$

which yields the stated bound. ∎

Theorem 1.3 (Robustness under camera placement error): Let $f : \mathbb{R}^d \to \mathbb{R}$ be modeled with a Gaussian process prior with squared–exponential kernel,

$$k(x, x') = \sigma_f^2 \exp\Big(- \tfrac{\|x-x'\|^2}{2\ell^2} \Big).$$

Suppose Bayesian optimization selects query viewpoints $x_t \in \mathbb{R}^d$, but the executed viewpoints are $\tilde{x}_t = x_t + \varepsilon_t$, with $\varepsilon_t \sim \mathcal{N}(0, \sigma_x^2 I_d)$ i.i.d. perturbations due to camera placement error. Define the effective objective $g(x) = \mathbb{E}[\, f(x + \varepsilon)\,]$.

Then g is governed by a GP with kernel,

$$k_g(x, x') \;=\; \sigma_f^2 \Big(\tfrac{\ell^2}{\ell^2 + 2\sigma_x^2} \Big)^{d/2} \exp\Big(- \tfrac{\|x-x'\|^2}{2(\ell^2+2\sigma_x^2)} \Big),$$

which is a squared-exponential kernel with effective hyperparameters, $\ell_{\text{eff}}^2 = \ell^2 + 2\sigma_x^2, \qquad \sigma_{f,\text{eff}}^2 = \sigma_f^2 \Big(\tfrac{\ell^2}{\ell^2+2\sigma_x^2} \Big)^{d/2}$.

Proof: Linearity of expectation preserves Gaussianity, so g is again a Gaussian process. Its covariance is.

$$k_g(x, x') \;=\; \mathbb{E}_{\varepsilon,\varepsilon'} \big[k(x + \varepsilon,\, x' + \varepsilon') \big],$$

where $\varepsilon, \varepsilon' \sim \mathcal{N}(0, \sigma_x^2 I_d)$ are independent. Let $\mu = x - x'$ and define $\delta = \varepsilon - \varepsilon'$. Then, $\delta \sim \mathcal{N}(0, 2\sigma_x^2 I_d)$, and hence,

$$k_g(x, x') \;=\; \sigma_f^2\, \mathbb{E}_\delta \Big[\exp\Big(- \tfrac{\|\mu+\delta\|^2}{2\ell^2} \Big) \Big].$$

This expectation can be evaluated using the Gaussian moment identity, $\mathbb{E}_{\delta\sim\mathcal{N}(0,\Sigma)} \Big[\exp\Big(- \tfrac{1}{2}(\mu+\delta)^\top A (\mu+\delta) \Big) \Big]$

$$= |I + \Sigma A|^{-1/2} \, \exp\Big(- \tfrac{1}{2} \mu^\top A (I + \Sigma A)^{-1} \mu \Big).$$

Substitute $A = \frac{1}{\ell^2} I_d$ and $\Sigma = 2\sigma_x^2 I_d$. Then,

$$|I + \Sigma A|^{-1/2} = \Big(\tfrac{\ell^2}{\ell^2+2\sigma_x^2} \Big)^{d/2}, A(I + \Sigma A)^{-1} = \tfrac{1}{\ell^2+2\sigma_x^2} I_d.$$

$$\implies k_g(x, x') = \sigma_f^2 \Big(\tfrac{\ell^2}{\ell^2+2\sigma_x^2} \Big)^{d/2} \exp\Big(- \tfrac{\|\mu\|^2}{2(\ell^2+2\sigma_x^2)} \Big)$$

∎

REFERENCES

[1] Soomin Lee, Le Chen, Jiahao Wang, Alexander Liniger, Suryansh Kumar, and Fisher Yu. Uncertainty guided policy for active robotic 3d reconstruction using neural radiance fields. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[2] Xuechao Zhang, Dong Wang, Sun Han, Weichuang Li, Bin Zhao, Zhigang Wang, Xiaoming Duan, Chongrong Fang, Xuelong Li, and Jianping He. Affordance-driven next-best-view planning for robotic grasping. In *Conference on Robot Learning (CoRL)*, 2023.

[3] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations (ICLR)*, 2017.

[4] Kunal Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. *Journal of Machine Learning Research*, 17(57), 2013.

[5] Atharva Gundawar, Som Sagar, and Ransalu Senanayake. Pac bench: Do foundation models understand prerequisites for executing manipulation policies? *arXiv preprint arXiv:2506.23725*, 2025.

[6] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[7] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017.

[8] Fereshteh Sadeghi and Sergey Levine. Cad2rl: Real single-image flight without a single real image. In *Robotics: Science and Systems (RSS)*, 2016.

[9] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object, 2023.

[10] Kyle Sargent, Zizhang Li, Tanmay Shah, Charles Herrmann, Hong-Xing Yu, Yunzhi Zhang, Eric Ryan Chan, Dmitry Lagun, Li Fei-Fei, Deqing Sun, and Jiajun Wu. Zeronvs: Zero-shot 360-degree view synthesis from a single image, 2024.

[11] Stephen Tian, Blake Wulfe, Kyle Sargent, Katherine Liu, Sergey Zakharov, Vitor Guizilini, and Jiajun Wu. View-invariant policy learning via zero-shot novel view synthesis, 2025.

[12] Lawrence Yunliang Chen, Chenfeng Xu, Karthik Dharmarajan, Muhammad Zubair Irshad, Richard Cheng, Kurt Keutzer, Masayoshi Tomizuka, Quan Vuong, and Ken Goldberg. Rovi-aug: Robot and viewpoint augmentation for cross-embodiment robot learning, 2024.

[13] Alexander Khazatsky, Karl Pertsch, Suraj Nair, and *et al.* DROID: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

[14] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations (ICLR)*, 2021.

[15] Dinesh Jayaraman and Kristen Grauman. Learning viewpoint-invariant visual representations by predicting views from novel viewpoints. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[16] Xiaoyu Wu and colleagues. Neural next-best-view planning for active 3d reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[17] Ying Lin and colleagues. Neural implicit active vision for scene understanding. In *International Conference on Computer Vision (ICCV)*, 2023.

[18] Harnaik Dhami, Vishnu D. Sharma, and Pratap Tokekar. Map-nbv: Multi-agent prediction-guided next-best-view planning for active 3d object reconstruction. *arXiv preprint arXiv:2307.04004*, 2023.

[19] Harnaik Dhami, Vishnu D. Sharma, and Pratap Tokekar. Pred-nbv: Prediction-guided next-best-view planning for 3d object reconstruction. *arXiv preprint arXiv:2307.04004*, 2023.

[20] Jingdong Hou, Han Zhang, and Xiaoming Liu. Learning to select views for efficient multi-view understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[21] Roman Marchant and Fabio Ramos. Bayesian optimisation for informative continuous path planning. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014.

[22] Roberto Calandra. Bayesian modeling for optimization and control in robotics. 2017.

[23] JT Wilson, R Moriconi, F Hutter, and MP Deisenroth. The reparameterization trick for acquisition functions. arxiv 2017. *arXiv preprint arXiv:1712.00424*, 2017.

[24] Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger. Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5), May 2012.

[25] Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Kevin Lin, Abhiram Maddukuri, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning, 2025.

[26] Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *arXiv preprint arXiv:2108.03298*, 2021.

[27] Michael Bain and Claude Sammut. A framework for behavioural cloning. In *Machine intelligence 15*, 1995.

[28] Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*. PMLR, 2019.

[29] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion, 2024.

[30] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.