Learning Factorized Diffusion Policies for Conditional Action Diffusion

Anonymous Author(s)
Affiliation
Address
email

Abstract: Diffusion models have emerged as a promising choice for learning robot skills from demonstrations. However, diffusion models are neither robust to visual distribution shifts nor sample-efficient for policy learning. In this work, we present 'Factorized Diffusion Policies' abbreviated as FDP, a novel theoretical framework to learn action diffusion models without the need to jointly condition on all observational modalities such as proprioception and vision. Using our factored approach leads to 10% absolute performance improvement for ten RLBench and four Adroit tasks when compared to a standard diffusion policy which jointly conditions on all modalities. Moreover, FDP results in 25% higher absolute performance across five RL Bench tasks with distribution shifts such as visual changes or distractors, where existing diffusion policies fail catastrophically. Our real-world experiments show that FDP is safe and relatively robust to deploy against visual distractors and appearance changes when compared to standard diffusion policies. Videos are available at https://fdp-policy.github.io/fdp-policy/.

1 Introduction

Diffusion models have emerged as a promising choice for learning robot skills from demonstrations [1]. Following various diffusion models, several generative models originally proposed in the vision literature have been used for robot learning, exploiting properties such as one-step inference [2, 3] and multimodal priors [4]. However, unlike computer vision, conditioning is critical in robotics due to the numerous observational modalities that influence the robot's action choices. Humans prioritize different sensory modalities according to the specific requirements of the task [5]. Humans have also been shown to prioritize the more reliable modality between vision and haptics [6]. Naturally, based on the task, robot skills should also depend more strongly on certain observational modes than others. For instance, repetitive motions like dance are more likely to depend on the robot's proprioception, while search and rescue is conditioned strongly on its vision.

However, the current method of training diffusion policies jointly conditions the action diffusion process on all the observational modalities for every task [1]. This is a monolithic joint conditioning approach – "when all you have is a hammer, everything looks like a nail". Learning the full conditional action distribution makes Diffusion Policies sensitive to distribution shifts in any of the modalities. We show that learning the full conditional results in low sample efficiency, brittleness to distribution shifts. In this work, we propose a novel *theoretical framework* 'Factorized Diffusion Policies' FDP for learning action diffusion models that decouples observational modalities for prioritization. At its core, FDP learns a *residual model* using some input modalities that have been omitted while training a base model with *prioritized inputs*. The base and residual model outputs are then composed to obtain samples from the full conditional action distribution. In addition, we present an architecture that enables efficient learning of the residual model in the FDP framework. We demonstrate that prioritization of modalities may yield significant gains in sample efficiency and naturally improves policy robustness to distribution shifts in the residual observations. Our contributions are as follows.

Submitted to the 9th Conference on Robot Learning (CoRL 2025). Do not distribute.

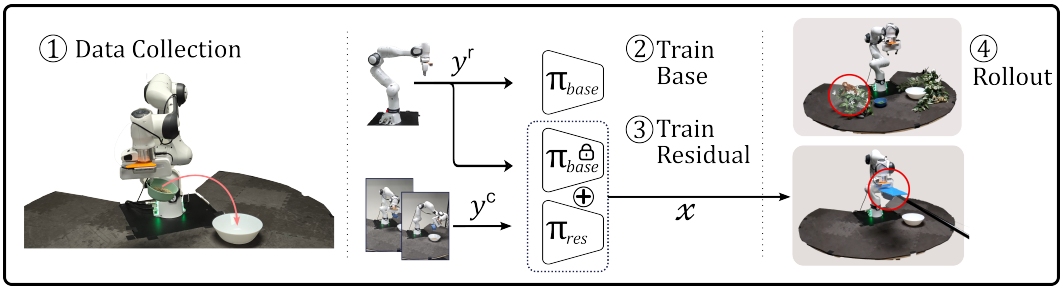

Figure 1: Training and inference for learning visuomotor policies using FDP with vision as a residual over proprioception. FDP is robust to deployment with distractors and camera occlusions.

1. We introduce Factorized Diffusion Policies (FDP), a novel theoretical framework for training diffusion models on robot demonstration data that decouples observation modalities for prioritization. We derive a novel loss function for learning a residual model on top of a policy trained with prioritized modalities, and propose an efficient architectural implementation to ease its learning.
2. Our experiments show that prioritization of observational modalities produces significant sample efficiency gains in several RLBench [7] and Adroit hand manipulation [8] environments. We show through several distractor experiments on RLBench that learning a visual residual model using FDP results in policies that are 25% more performant over standard diffusion policies.
3. We collect demonstrations across several task environments on a real robot and evaluate both FDP and standard diffusion policy in the original environments as well as in modified versions with visual distractors and appearance changes. In our real-world experiments, FDP outperforms diffusion policies by over 40% in the presence of distractors, occlusions and appearance changes.

2 Background and Related Work

Diffusion Models. Gaussian diffusion models [9] learn the reverse diffusion kernel $p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ for a fixed forward kernel that adds Gaussian noise at each step $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}(\boldsymbol{x}_t; \sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\mathcal{I})$, such that $q(\boldsymbol{x}_T) \approx \mathcal{N}(0, \mathcal{I})$. Here, $t <= T$ is the diffusion time step and α_t is the noise schedule. For training the model, maximization of the evidence lower bound on the log-likelihood of the data distribution $\log q(\boldsymbol{x}_0)$ yields the commonly used loss function in Equation 1 [10, 11].

$$\mathcal{L}_t(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0),\, \boldsymbol{\epsilon}_0 \sim \mathcal{N}(0,\mathcal{I})} \left[\lambda_t[||\boldsymbol{\epsilon}_0 - \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)||_2^2]\right] \tag{1}$$

Here, λ_t, a function of α_t is the weighting parameter for different time steps, usually taken as 1 [10]. The model is trained to predict the noise $\boldsymbol{\epsilon}_0$ added to the data sample $\boldsymbol{x}_0$ to generate the noisy sample $\boldsymbol{x}_t$ taken as input to the network.

Connection to Score-based Models. Song et al. [12] presented a unified framework showing that both diffusion models [9, 10] and score-based models [13] can be interpreted as discretizations of different forward stochastic differential equations (SDEs). Denoising score matching (DSM) [14] is used to learn the score $\nabla_{\tilde{\boldsymbol{x}}} \log q_\sigma(\tilde{\boldsymbol{x}})$ at different noise scales σ required for sampling from the data distribution via the corresponding reverse-time SDEs [15]. Explicit Score Matching (ESM) [16, 14] was proposed to estimate this score by minimizing the Fisher divergence with the Gaussian-smoothed data distribution $q_\sigma(\tilde{\boldsymbol{x}}) = \int q(\boldsymbol{x})\mathcal{N}(\tilde{\boldsymbol{x}}; \boldsymbol{x}, \sigma^2 I) d\boldsymbol{x}$. DSM alleviates the computational difficulties of ESM [14, 17, 18], and is shown in Equation 2, where $s_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}})$ represents the learned score model.

$$\mathcal{J}_{\sigma_t}(\boldsymbol{\theta}) \overset{ESM}{=} \mathbb{E}_{q_{\sigma_t}(\tilde{\boldsymbol{x}})}\left[\frac{1}{2}||\nabla_{\tilde{\boldsymbol{x}}} \log q_{\sigma_t}(\tilde{\boldsymbol{x}}) - s_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}})||_2^2\right] \overset{DSM}{=} \mathbb{E}_{q_{\sigma_t}(\boldsymbol{x},\tilde{\boldsymbol{x}})}\left[\frac{1}{2}||\nabla_{\tilde{\boldsymbol{x}}} \log q_{\sigma_t}(\tilde{\boldsymbol{x}}|\boldsymbol{x}) - s_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}})||_2^2\right] + C \tag{2}$$

Diffusion models use a forward transition kernel $q(\tilde{\boldsymbol{x}}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}_t; \sqrt{\bar{\alpha}_t}\boldsymbol{x}_{t-1}, (1-\bar{\alpha}_t)I)$ with discrete time and $\bar{\alpha}_i = \prod_{j=1}^{i} \alpha_j$, yielding the loss shown in Equation 1, while score-based model typically use $\mathcal{N}(\boldsymbol{x}_t; \boldsymbol{x}_{t-1}, \sigma_t^2 I)$, where α_t and σ_t are respective noise scales. Based on the equivalence of Equations 1 and 2, an optimal diffusion model learned using Equation 1, is related to the score of

the diffused data distribution by $\epsilon^*_{\theta}(\boldsymbol{x}_t, t)/\sqrt{1-\bar{\alpha}_t} = \nabla_{\boldsymbol{x}_t} \log q(\boldsymbol{x}_t)$ [12, 11]. Typically, diffusion models generate samples via progressive denoising through the reverse diffusion process [10], while score-matching models sample from the data distribution using Langevin dynamics [19, 20].

Relevant work in Robotics. Sample efficiency and generalization are of primary importance in robotics, as scaling the collection of multimodal data is difficult and the number of variations of tasks is unbounded. While generative model families such as diffusion [1], score-based models [21], stochastic interpolants [4], and flows [2] have been applied in robotics, they do not address these limitations. Prior compositional works have tried to address these problems by composing learned constraints to generalize to new task combinations in manipulation [22] and planning [23], or composing distributions across heterogeneous modalities for tool use [24]. However, all the previous works compose learned or analytical distributions, limiting their application to combinations of existing solutions. Instead, in our FDP framework, we learn a residual over a base policy that, when composed with the base policy, provides samples corresponding to the data distribution. Recent augmentation-based methods [25, 26] improve generalization but add a substantial computational overhead and remain vulnerable to visual failures like temporary camera occlusions or dynamic scene changes. In contrast, FDP is an algorithmic improvement that achieves robustness to such perturbations without data augmentations as demonstrated in our real-world experiments.

3 Methodology

Assume that we have robot demonstrations $D = \{(\boldsymbol{x}, \boldsymbol{y})_i\}$ where $i = 1..N$, consisting of actions $\boldsymbol{x}$ and different observational modalities $\boldsymbol{y}^{1:M}$, such as images or point clouds from different cameras and proprioception data. We are interested in learning $p(\boldsymbol{x}|\boldsymbol{y})$ from the data such that given a task description, current camera images, state of the robot, and other observations, we can sample an action $\boldsymbol{x}$ with a high likelihood in the data distribution. Most treatments of diffusion models have been studied primarily in the context of single-modality distributions, such as those over image pixels [10, 27, 13]. This formulation has been directly adopted by the robotics community [1, 21, 28], leading to the popular optimization objective shown in Equation 3.

$$\mathcal{L}_t(\boldsymbol{\theta}) = \mathbb{E}_{(\boldsymbol{x}_0, \boldsymbol{y}) \sim q(\boldsymbol{x}_0, \boldsymbol{y}),\, \boldsymbol{\epsilon}_0 \sim \mathcal{N}(0, \mathcal{I})} \left[||\boldsymbol{\epsilon}_0 - \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{y}, t)||_2^2 \right] \tag{3}$$

Here, the network $\boldsymbol{\epsilon}_\theta$ is conditioned on the observation $\boldsymbol{y}$, and is trained to predict the noise added to the action sample $\boldsymbol{x}$. Although prior work in robotics adopts this conditional formulation [1], it is assumed without formal justification that the trained network maximizes the log-likelihood of the conditional distribution $p(\boldsymbol{x}|\boldsymbol{y})$. Hence, we present our first result as follows.

Lemma 3.1. *The diffusion loss function $\mathcal{L}_t(\boldsymbol{\theta})$ as defined in Equation 3, in expectation over the time-steps $1 \leq t \leq T$, maximizes the variational lower bound on the log-likelihood of the conditional data distribution $\log q(\boldsymbol{x}|\boldsymbol{y})$, under a Markovian noising process $\hat{q}(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ and the conditional reverse transition kernel as $\hat{q}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{y})$.*

The proof for Lemma 3.1 is presented in Appendix C.2. In Equation 3, $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{y}, t)$ arises from the reparametrization of the reverse transition kernel $q_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{y}^{1:M})$, and from a score-based perspective, it learns the score of the full action conditional $\nabla_{\boldsymbol{x}_t} \log q(\boldsymbol{x}_t|\boldsymbol{y}^{1:M})$ times a constant. We argue that learning the full conditional directly is restrictive in several aspects of robot learning. Firstly, it necessitates the joint collection of the robot action and all observational modalities. This restriction makes it impossible to add observational modalities later. Secondly, the model is vulnerable to even small distribution shifts in *any* modality. These shifts require a prohibitively large amount of data to address when the observation modalities are high-dimensional. Finally, among the multiple observation modalities it is hard to pinpoint the level of each mode's task dependent influence with limited data. Hence, we present FDP, a method to add structure and decouple observational modalities in the score of the full action conditional $\nabla_{\boldsymbol{x}_t} \log q(\boldsymbol{x}_t|\boldsymbol{y}^{1:M})$. By factorizing modalities using Bayes' theorem and learning residuals for subsequent terms, FDP effectively encodes task requirements and learns policies robust to distribution shifts in residual modalities.

3.1 Factorized Diffusion Policies

Let $\boldsymbol{y}^{1:k}$ be the prioritized observational modalities of the M total modalities, where $\boldsymbol{y}^{1:k} \equiv \boldsymbol{y}^1, .., \boldsymbol{y}^k$ and $1 \leq k < M$. To decouple the observational modalities, we utilize Bayes' theorem on the score of the full action conditional to obtain the following.

$$\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t|\boldsymbol{y}^{1:M}; \boldsymbol{\theta}, \boldsymbol{\phi}) = \nabla_{x_t} \log p(\boldsymbol{x}_t|\boldsymbol{y}^{1:k}; \boldsymbol{\theta}) + \nabla_{x_t} \log p(\boldsymbol{y}^{k+1:M}|\boldsymbol{x}_t, \boldsymbol{y}^{1:k}; \boldsymbol{\phi}) \tag{4}$$

To prioritize modalities $y^{1:k}$, we propose to first learn a diffusion policy π_{base}: $\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t, \boldsymbol{y}^{1:k}, t)$ that corresponds to the first score term on the right-hand side of Equation 4. To learn the second score term, explicitly training a classifier $p(\boldsymbol{y}^{k+1:M}|\boldsymbol{x}_t, \boldsymbol{y}^{1:k})$ [29] is impractical due to the high dimensionality and continuity of observational modalities $\boldsymbol{y}^{1:M}$, such as images. Hence, we employ explicit score matching [16, 14] as shown in Equation 5.

$$D_F^t = \mathbb{E}_{p_{\alpha,\tau}(\boldsymbol{x}_t, \tilde{\boldsymbol{y}}^{1:M})}\left[\frac{1}{2}||\nabla_{\boldsymbol{x}_t} log\ p_{\boldsymbol{\phi}}(\tilde{\boldsymbol{y}}^{k+1:M}|\boldsymbol{x}_t, \tilde{\boldsymbol{y}}^{1:k}) - \nabla_{\boldsymbol{x}_t} log\ p_{\alpha,\tau}(\tilde{\boldsymbol{y}}^{k+1:M}|\boldsymbol{x}_t, \tilde{\boldsymbol{y}}^{1:k})||_2^2\right] \tag{5}$$

Here, observational modalities $\boldsymbol{y}^{1:M}$ can be noised with a Gaussian kernel $\mathcal{N}(\tilde{\boldsymbol{y}}; \boldsymbol{y}, \tau^2 I)$ of variance τ^2 that is small enough such that $p_\tau(\tilde{\boldsymbol{y}}^i) \approx p(\boldsymbol{y}^i)$. Chao et al. [18] show that the empirical score is difficult to estimate for large datasets and derive the denoising likelihood score matching (DLSM) objective for conditional distributions, which forms the basis for our next result.

Theorem 3.2. *Explicit score matching for $\nabla_{\boldsymbol{x}_t} log\ p_{\boldsymbol{\phi}}(\tilde{\boldsymbol{y}}^{k+1:M}|\boldsymbol{x}_t, \tilde{\boldsymbol{y}}^{1:k})$, as expressed in Equation 5 with $\boldsymbol{x}$ is noised with the diffusion transition kernel $\mathcal{N}(\boldsymbol{x}_t; \sqrt{\alpha_t}\boldsymbol{x}, (1-\alpha_t)\mathcal{I})$, is equal up to a constant to the following loss:*

$$L_{res}^t(\boldsymbol{\phi}) = \mathbb{E}_{p_\tau(\boldsymbol{x}_0, \boldsymbol{y}^{1:M}, \tilde{\boldsymbol{y}}^{1:M})}\mathbb{E}_{\boldsymbol{\epsilon}_0 \sim \mathcal{N}(0,\mathcal{I})}\left[\frac{1}{2}||\boldsymbol{\epsilon}_0 - \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, \tilde{\boldsymbol{y}}^{1:k}, t) - \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, \tilde{\boldsymbol{y}}^{1:M}, t)||_2^2\right] \tag{6}$$

The proof for Theorem 3.2 is presented in Appendix C.3. Equation 6 allows us to train the score of the classifier π_{res}: $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, \tilde{\boldsymbol{y}}^{1:M}, t)$ as a residual over frozen π_{base} to predict noise ϵ_0 added to the action x_0. Learning π_{res} as a residual of π_{base} ensures that the model does not overfit modalities $y_{k+1:M}$, but only learns correlations to bridge the gap between the expected score and the predicted score of the model π_{base} trained on the prioritized modalities $y^{1:k}$. Hence, policies learned in this factorized way are naturally robust to distribution shifts in the residual modalities. Moreover, explicit prioritization of $y^{1:k}$ by training π_{base} prior to learning the residual leads to sample efficiency, as the model learns correlations with the stronger modality without having to attend to other modalities. Since diffusion models are trained on discrete time steps, the residual is learned on the same time discretization as used for π_{base}. Once trained, actions can be sampled from the conditional distribution $p(\boldsymbol{x}|\boldsymbol{y}^{1:M})$ using reverse diffusion [10] on the composition [30] of π_{base} and π_{res}:

$$\boldsymbol{x}_{t-1} \sim \mathcal{N}\left(\boldsymbol{x}_t; \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}(\boldsymbol{x}_t, \boldsymbol{y}^{1:M}, t)\right), \sqrt{1-\alpha_t}\mathcal{I}\right) \tag{7}$$

$$\boldsymbol{\epsilon}(\boldsymbol{x}_t, \boldsymbol{y}^{1:M}, t) = \boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t, \boldsymbol{y}^{1:k}, t) + \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, \boldsymbol{y}^{1:M}, t) \tag{8}$$

The specific instantiations of FDP for combinations of modalities are presented in Appendix C.1. Note that the base mode π_{base}: $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{y}^{1:k}, t)$ can be further decomposed with respect to observational modalities. In this work, the residual model π_{res} is learned in expectation over data collected jointly for all modalities $y^{1:M}$, of which $y^{1:k}$ are used for training π_{base}. However, we emphasize that an important feature of our learning formulation is that it enables decoupled data collection for additional conditionals that could then be used to learn the residual. This potentially may alleviate some difficulties encountered for scaling coupled data in robotics and is left for future work.

3.2 Architectural Implementations of FDP

The base and residual models in FDP, denoted by π_{base} and π_{res}, can be instantiated using standard architectures such as UNet [31] or DiT [32]. FDP involves the additional step of learning π_{res} as a residual over a frozen π_{base}. During inference we compose the outputs of these as shown in Figure 2 [b]. However, we find this late-stage residual learning to be inefficient in practice and propose a more integrated way to compose π_{base} and π_{res}, as shown in Figure 2. This architecture enables a simplified training objective for the residual model, equivalent to Equation 3. Instead of

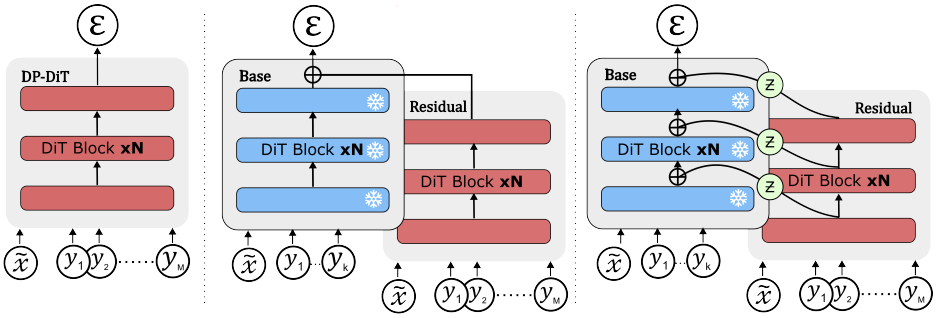

Figure 2: Architectural representations for [a] diffusion policy that jointly conditions on all observational modalities, [b] simple FDP architecture that composes the score outputs from π_{base} and π_{res} and [c] FDP architecture with block-wise composition with a zero layer $\mathcal{Z}$ applied on π_{res}.

learning a residual for the final score output, π_{res} learns the blockwise residual over the intermediate outputs of the frozen π_{base}. Specifically, let $\mathcal{F}^i_{\text{base}}$ and $\mathcal{F}^i_{\text{res}}$ denote the i-th DiT block outputs of the base and residual models, respectively. Then the composed output at level i can be written as $\mathcal{F}^i_{base}(\boldsymbol{x}', \boldsymbol{y}'^{1:k}) + \mathcal{Z}(\mathcal{F}^i_{res}(\boldsymbol{x}', \boldsymbol{y}'^{1:M}))$, where x' and $y'^{1:M}$ are layer inputs. Similar to Zhang et al. [33], $\mathcal{Z}$ is a zero-initialized layer to avoid harmful updates at the start of the training and to ensure that gradient updates to the residual model improve the predictions of the composed model over π_{base}. Crucially, we find that preserving the diversity of π_{base} is essential: overfitting the base model leaves little residual signal to learn, reducing generalization. Our experiments show that selecting the π_{base} checkpoint with the lowest validation loss provides a good foundation for residual learning. Our residual model is structured following the Vision Transformers architecture [34]. In π_{res}, all observational modalities are passed through self-attention layers after encoding. Our visual residual model encodes camera images into a single patch to reduce computational overhead. Complete implementation details and architectural ablations are provided in Appendix D and H respectively.

4 Simulation Experiments

We train and evaluate FDP and related baselines in ten tasks of RLBench [7] and four tasks of Adroit [35] and Robomimic [36] each. RLBench is a large-scale simulation benchmark for robotic manipulation, offering multiple sensory modalities. For our experiments, we use several predefined single-task environments, and train policies on data collected using an in-built-planner with joint states as the action space. To study the impact of distribution shifts in observational modalities, we also evaluate the trained policies on six modified RLBench environments that introduce appearance changes and visual distractors. Adroit [35, 8] is a simulated 24-DoF anthropomorphic hand platform widely used for dexterous manipulation research. Each task provides a low-dimensional hand-crafted environment state alongside rich proprioceptive feedback, which is used to train policies for dynamic manipulation. RoboMimic [36] provides human demonstration data collected across four RoboSuite [37] environments, featuring tasks with varying levels of precision and horizon length. More details in the Appendix E and our webpage `https://fdp-policy.github.io/fdp-policy/`.

Baselines. For evaluation of sample efficiency in visuomotor tasks, we compare against several approaches that differ in the way in which they probabilistically model generative policy learning. However, for all approaches, we choose DiT-small (∼90M) [32] as our model architecture. We implement Diffusion Policy [1] using DiT, referred to as DP-DiT in our results. For comparison, we also include UNet [31] implemented by Chi et al. [1] in our baselines as DP-UNet. We reformulate POCO [24] to compose the modalities of proprioception y^r and vision y^c. We train the motion π_{base} and vision π_{res} models independently, prior to sampling from the composed distribution [30] using $\boldsymbol{\epsilon}(\boldsymbol{x}_t, \boldsymbol{y}^r, \boldsymbol{y}^c, t) = \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, \boldsymbol{y}^r, \boldsymbol{y}^c, t) + \lambda * \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{y}^r, t)$. Here, $\lambda = 0.1$ based on POCO's ablations [24]. We also report results for classifier-free guidance [38] as CFG, where we train a single model and

switch out the vision modality with a probability of 0.2. We then sample using $\epsilon(\boldsymbol{x}_t, \boldsymbol{y}^r, \boldsymbol{y}^c, t) = \lambda_1 * \hat{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{y}^r, \boldsymbol{y}^c, t) + \lambda_2 * \hat{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{y}^r, \phi, t)$, where we set $\lambda_1 = 1.1$ and $\lambda_2 = 0.1$, as suggested by [38]. For real-world and distractor experiments in simulation, we compare against DP-DiT.

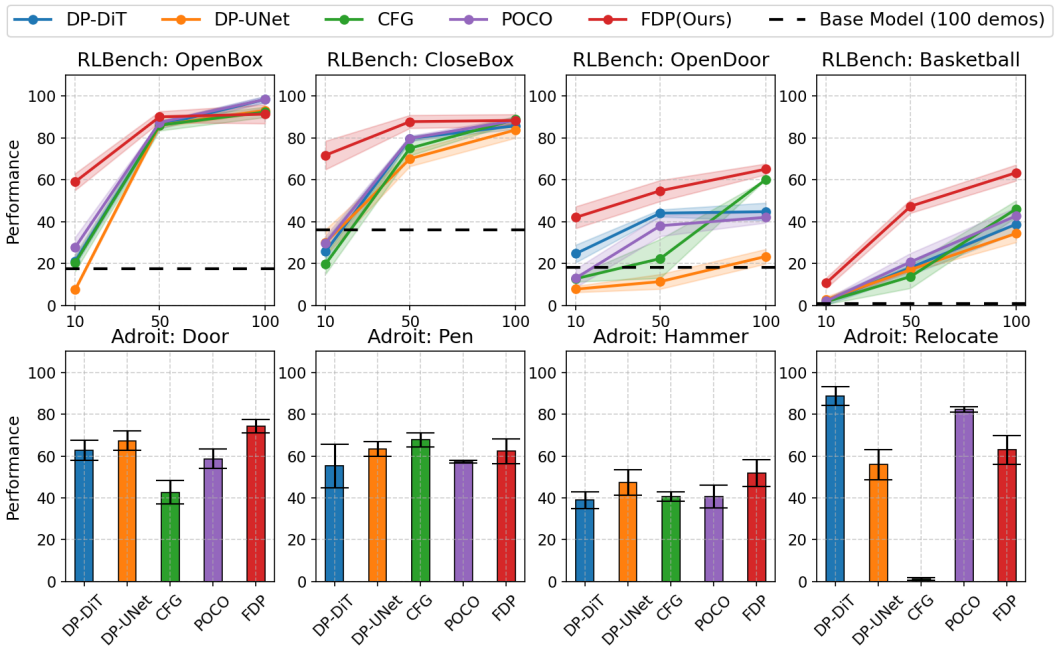

Figure 3: Evaluation for FDP and baselines on four RLBench and four Adroit tasks. FDP results in sample efficient policies at low number of demonstrations. More plots in Appendix F.

Research Question 1: Can task-specific prioritization of modalities using FDP lead to sample efficiency gains in learning visuomotor tasks? Prioritization of proprioception using the FDP framework outperforms all baselines in four tasks each of RLBench and Adroit, across different number of demonstrations as shown in Figure 3. In RLBench, FDP achieves 20% higher performance on average with 10 demonstrations and 10% higher performance on average with 100 demonstrations over the strongest baseline. FDP results in sample-efficient policies, especially with low number of demonstrations as the model is able to attend strongly to proprioception, only learning a residual for the visual observations. Note that a fully trained π_{base} motion model on 100 demonstrations fails in these tasks in isolation, implying that vision is required to complete these tasks successfully. In the Adroit environments, proprioception prioritized FDP performs better by $\sim$10% on average over DP-DiT for Door, Pen and Hammer tasks. For the Relocate task, which involves grasping a ball placed randomly on the table and relocating it to a random goal location, the robot action is strongly influenced by the environment state specifying the ball and the goal location. Hence, learning a motion π_{base} does not work in favor of improving policy performance, as the effect of the state of the environment is learned as a residual over π_{base}. Prioritization of proprioception will lead to sample-efficiency in repetitive tasks. Tasks that correlate heavily with robot proprioception are not uncommon as the robot is solving them in the first person view, and can move close to the object if required. More results on these and six other RLBench tasks in Appendix F.

Research Question 2: Does learning the visual modality as a residual in the FDP framework result in robustness to distractors and appearance changes? We present the results of the policy evaluations in the distractor environments in Table 1.1. Both DP-DiT and FDP are trained on 100 demonstrations collected in the original environment and evaluated in three settings: the original environment, an environment augmented with distractors, and an environment with visual modifications to the manipulated objects. FDP significantly outperforms DP-DiT in both distractor-augmented and visually modified environments by more than 40%. Additionally, we further collect

five demonstrations in each modified environment to investigate the benefits of few-shot adaptation to out-of-distribution data. Notably, FDP responds more effectively to additional demonstrations in the modified settings, improving it's performance by 15% on average over DP's 10%. In particular, FDP updates only the residual model π_{res} with new demonstrations, adjusting the conditional distribution on visual modalities $p(y^c|x, y^r)$ without modifying the full conditional action distribution $p(x|y^r, y^c)$. We extend this setting to point clouds and learn a visual residual on DP3 [39], as compared to DP3 with RGB inputs. Point clouds are sample-efficient for policy learning as they effectively encode the geometric structure of the scene in a single modality [39, 40, 41]. However, our distractor experiments show that FDP with a visual residual learned over DP3 is $\sim 20\%$ more performant than DP3 with RGB inputs. Further experimental details are in Appendix F.

Table 1.1: Robustness to Visual Distractors (100 demos)
(FDP significantly improves generalization to visual changes)

Task	Environment	DP-DiT	FDP
OpenBox	Original	**98.3** ± 1.5	91.3 ± 4.5
	Zero-shot color	43.3 ± 2.5	**46.7**± 1.5
	5 demos color	34.7 ± 3.5	**76.7** ± 0.6
	Zero-shot distractors	1.7 ± 2.1	**16.7**± 2.1
	5 demos distractors	42.3 ± 4.0	**53.3** ± 2.3
Basketball in Hoop	Original	38.7 ± 4.2	**63.3** ± 3.8
	Zero-shot color	29.0 ± 8.7	**63.0** ± 1.0
	5 demos color	13.0 ± 0.0	**45.0** ± 2.6
	Zero-shot distractors	0.7 ± 1.2	**56.3** ± 3.2
	5 demos distractors	2.7 ± 1.2	**39.7** ± 4.9
Open Door	Original	44.7 ± 4.2	**65.0** ± 2.6
	Zero-shot color1	0.0 ± 0.0	**14.3** ± 3.1
	5 demos color1	17.7 ± 3.1	**52.0** ± 7.2
	Zero-shot color2	0.3 ± 0.6	**30.7** ± 2.5
	5 demos color2	20.0 ± 5.2	**53.7** ± 3.8

Table 1.2: Block Pick Success Rates
(FDP performs better in tasks with less variation.)

Variations	Model	10 demos	50 demos	Distractors
Small	FDP	**73.7** ± 3.8	**98.7** ± 1.5	**99.3** ± 0.6
	DP-DiT	29.7 ± 3.1	95.3 ± 3.2	0.0 ± 0.0
Medium	FDP	**21.3** ± 3.5	55.0 ± 2.6	**58.3** ± 3.1
	DP-DiT	12.0 ± 1.0	**69.0** ± 7.0	2.0 ± 1.0
Large	FDP	**6.3** ± 3.1	20.3 ± 3.5	0.7 ± 1.2
	DP-DiT	3.3 ± 0.6	**45.7** ± 7.1	0.0 ± 0.0

Table 1.3: Robomimic Lowdim Task Success Rates (100 demos)
(Evaluating FDP at precise and long-horizon manipulation.)

Task	DP-DiT	CFG	POCO	FDP
Lift	99.0 ± 1.7	98.7 ± 0.6	98.7 ± 1.5	99.7 ± 0.6
Can	99.0 ± 1.0	98.7 ± 1.5	98.7 ± 1.5	99.7 ± 0.6
Square	**80.3** ± 4.6	**80.0** ± 3.0	76.3 ± 6.0	58.0 ± 6.6
Toolhang	**60.0** ± 7.5	**60.7** ± 3.5	55.7 ± 7.5	45.7 ± 3.8

Table 1: Tests for robustness and the effects of factorization across domains.

Research Question 3: How sensitive is the task performance to prioritization perception for visuomotor tasks? In visuomotor tasks, prioritizing proprioception over learning the full conditional distribution can be advantageous when the object placement diversity is low or when robustness to visual distribution shifts is critical. To demonstrate this, we construct three RLBench environments for a block-picking task, each featuring an increasingly larger object placement area. The results are presented in Table 1.2. As expected, FDP is sample-efficient and outperforms DP-DiT across all variation scales with only 10 demonstrations. However, with increasing number of demonstrations, DP-DiT eventually surpasses FDP at larger variation scales. Notably, DP-DiT still fails when visual distractors are introduced, whereas FDP remains robust and outperforms DP-DiT even at higher scales of task-variation in distractor-augmented environments. We also evaluated the performance of DP-DiT and FDP for fine-manipulation tasks on the Robomimic dataset. Although FDP is more sample efficient for the tasks of `Lift` and `Can`, it achieves a lower success rate than DP-DiT for `Square` and `Toolhang`, as shown in Table 1.3. This is to be expected, as fine-manipulation tasks present a bottleneck in the joint state-action distribution, and FDP factorizes the distribution into components where some modalities are learned as residuals over the others.

5 Real-world Experiments

We evaluate FDP and the DP-DiT baseline across four real-world domains and report their task success rates. The domains are – *Close Drawer* as a simple task where the robot has to push the drawer; *Put Block in Bowl* that assesses the policy's ability to perform precise pick-and-place actions; *Pour in Bowl* to evaluate the policy's dexterity in operating near joint limits and *Fold Towel* to assess effectiveness in manipulating deformable objects.

We collect 50 demonstrations per domain on a Franka FR3 robot using a 6D space mouse, recording both proprioceptive and visual observations from two cameras—one mounted on the gripper and a

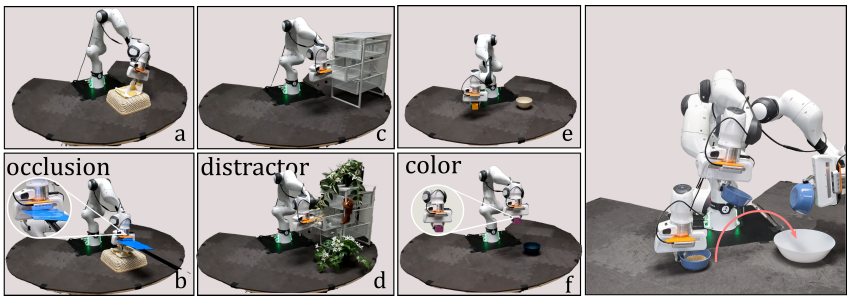

Figure 4: Task domains and their variations. In `occlusion`, the visual input is blocked using a board; `distractor`, flower pots and toys are introduced into the scene; and in `color`, the color of the manipulated object is altered during evaluation.

static camera covering the workspace. The trained policies are evaluated on four task variations in each domain: (a) `default`: an in-distribution setup matching the conditions used during demonstration collection; (b) `color`: the object's color is altered to test robustness to visual appearance changes; (c) `distractor`: novel, unseen objects such as vegetation props and soft toys are added to the scene to introduce clutter; and (d) `occlusion`: visual input is intermittently blocked during policy rollout to simulate partial observability. Figure 4 shows different task domains and their variations used in our experiments. More details on the robot system setup can be found in Appendix G. We use 10 rollouts in each experiment and report the task success rate as shown in Table 2.

Task Domain	`default`		`color`		`distractor`		`occlusion`	
	DP	**FDP**	**DP**	**FDP**	**DP**	**FDP**	**DP**	**FDP**
Close Drawer	**90**	**90**	**90**	**90**	10	**80**	0	**80**
Put Block in Bowl	**80**	**80**	0	**60**	0	**60**	10	**60**
Pour in Bowl	70	**80**	40	**80**	20	**60**	10	**50**
Fold Towel	40	**60**	40	**70**	30	**70**	10	**50**

Table 2: Success rates (%) of Diffusion Policies (DP) and Factorized Diffusion Policies (FDP) across real-world tasks with 10 rollouts for each condition.

Result Analysis. We find that FDP is robust to distribution shifts in the environment. DP regularly produces unachievable robot actions under `distractor` and `occlusion` settings, often triggering safety stops, resulting in task failure. In contrast, FDP guided by its motion prior, consistently generates stable actions even under severe occlusions and cluttered scenes, yielding an average absolute performance improvement of 40% over DP. In the `default` experiment we observe that the FDP policy outperforms DP in the pouring and towel-folding tasks, which require precise object manipulation. With just 50 demonstrations, DP overfits in these fine-grained tasks due to limited motion diversity, whereas FDP, leverages its residual guidance and effectively learns robust policies.

6 Conclusion

We present Factorized Diffusion Policies (FDP), a novel theoretical framework for prioritization of observation modalities in policy learning. We provide probabilistic grounding for diffusion policy learning and reveal the pitfalls of learning a full conditional on all the observational modalities. FDP decouples the modalities and proposes a framework for their selective prioritization. We derive a novel loss function to realize the decoupling of modalities and support it with a novel architecture for efficient training. Through extensive experiments, we demonstrate several benefits of modality prioritization, including improved sample efficiency and increased robustness to visual distractors and camera occlusions when learning a residual for vision. FDP opens new avenues for future research, such as scalable integration of diversely sourced observational modalities for robot policy learning. Finally, our real-world experiments highlight that FDP maintains strong performance even under significant visual disruptions, outperforming diffusion policies by over 40%.

7 Limitations

FDP is a theoretical framework to decouple observational modalities for robot policy learning. Predominantly, we present results for visuomotor tasks, but our method is generic and can be extended to other modalities. We see the following issues with our FDP framework –

7.1 Prioritizing Modalities

We focus on the benefits and pitfalls of prioritizing proprioception and alternatively learning a residual for vision in our experiments. However, for applications in a broader scope, the choice of modalities to be prioritized will need to be studied and is not answered in this work. Understanding which modality to prioritize for a particular task can be a challenging question for diffusion models in general and might be severly task dependent. This might indicate that there might be an inference time choice of composing modality that an agent might have to make.

7.2 Factorizing modalities is not a general solution for all tasks

As we show in our experiments, decoupling modalities may not be the right choice for every task or skill. This is because some tasks require the full joint distribution of observations. We also want to point out here that it is challenging to know whether a task requires the full joint distribution or the factored distribution in a specific prioritized order. Future approaches could learn to automatically attend on the right modality or the joint distribution, much like humans do.

7.3 Architectural Choices

Moreover, the framework can also benefit from further architectural improvements that realize a better trade-off between the strength of the guidance imparted by the residual model π_{res} and its robustness to perturbations in its inputs. The current training setup requires a two-step process for learning π_{base} and π_{res} that presents a time and computational overhead over training standard diffusion policies which might be cumbersome at deployment.

7.4 Baselines outside of Diffusion based policy models

In this work we only compare to diffusion based policy models as we are attempting to improve their robustness and extend their capabilities of factorization. However, a large scale comparison against different type of policy models is left to be done. For now we do not think comparisons against non-diffusion policy types is critical but it is desirable to understand when to use which type of policy for a robot.

7.5 Large Vision Action Models

There are large scale vision action models that can perform tasks specified by language in visual environments sometimes even zero shot. Here we are studying how to learn individual skills using diffusion based behavior cloning approaches. These skills might then be used in a larger stack of a vision-action model. The question of sample efficiency and robustness to distractors will always be important independent of the scale of the models themselves.

References

[1] 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. The International Journal of Robotics Research, page 02783649241273668, 2023.

[2] Q. Zhang, Z. Liu, H. Fan, G. Liu, B. Zeng, and S. Liu. Flowpolicy: Enabling fast and robust 3d flow-based policy via consistency flow matching for robot manipulation. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 39, pages 14754–14762, 2025.

[3] A. Prasad, K. Lin, J. Wu, L. Zhou, and J. Bohg. Consistency policy: Accelerated visuomotor policies via consistency distillation. arXiv preprint arXiv:2405.07503, 2024.

[4] K. Chen, E. Lim, K. Lin, Y. Chen, and H. Soh. Don't start from scratch: Behavioral refinement via interpolant-based policy diffusion. arXiv preprint arXiv:2402.16075, 2024.

[5] B. Wahn and P. König. Is attentional resource allocation across sensory modalities task-dependent? Advances in cognitive psychology, 13(1):83, 2017.

[6] M. O. Ernst and M. S. Banks. Humans integrate visual and haptic information in a statistically optimal fashion. Nature, 415(6870):429–433, 2002.

[7] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison. Rlbench: The robot learning benchmark & learning environment. CoRR, abs/1909.12271, 2019. URL http://arxiv.org/abs/1909.12271.

[8] J. Fu, A. Kumar, O. Nachum, G. Tucker, and S. Levine. D4rl: Datasets for deep data-driven reinforcement learning, 2020.

[9] J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015.

[10] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models, 2020. URL https://arxiv.org/abs/2006.11239.

[11] C. Luo. Understanding diffusion models: A unified perspective, 2022. URL https://arxiv.org/abs/2208.11970.

[12] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. arXiv preprint arXiv:2011.13456, 2020.

[13] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution, 2020.

[14] P. Vincent. A connection between score matching and denoising autoencoders. Neural computation, 23(7):1661–1674, 2011.

[15] B. D. Anderson. Reverse-time diffusion equation models. Stochastic Processes and their Applications, 12(3):313–326, 1982.

[16] A. Hyvärinen and P. Dayan. Estimation of non-normalized statistical models by score matching. Journal of Machine Learning Research, 6(4), 2005.

[17] Y. Song and D. P. Kingma. How to train your energy-based models, 2021.

[18] C.-H. Chao, W.-F. Sun, B.-W. Cheng, Y.-C. Lo, C.-C. Chang, Y.-L. Liu, Y.-L. Chang, C.-P. Chen, and C.-Y. Lee. Denoising likelihood score matching for conditional score-based data generation, 2022. URL https://arxiv.org/abs/2203.14206.

[19] G. O. Roberts and R. L. Tweedie. Exponential convergence of langevin distributions and their discrete approximations. 1996.

[20] G. O. Roberts and J. S. Rosenthal. Optimal scaling of discrete approximations to langevin diffusions. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 60(1): 255–268, 1998.

[21] M. Reuss, M. Li, X. Jia, and R. Lioutikov. Goal-conditioned imitation learning using score-based diffusion policies. arXiv preprint arXiv:2304.02532, 2023.

[22] W. Liu, J. Mao, J. Hsu, T. Hermans, A. Garg, and J. Wu. Composable part-based manipulation. arXiv preprint arXiv:2405.05876, 2024.

[23] Z. Yang, J. Mao, Y. Du, J. Wu, J. B. Tenenbaum, T. Lozano-Pérez, and L. P. Kaelbling. Compositional diffusion-based continuous constraint solvers. arXiv preprint arXiv:2309.00966, 2023.

[24] L. Wang, J. Zhao, Y. Du, E. H. Adelson, and R. Tedrake. Poco: Policy composition from and for heterogeneous robot learning. arXiv preprint arXiv:2402.02511, 2024.

[25] T. Yu, T. Xiao, A. Stone, J. Tompson, A. Brohan, S. Wang, J. Singh, C. Tan, J. Peralta, B. Ichter, et al. Scaling robot learning with semantically imagined experience. arXiv preprint arXiv:2302.11550, 2023.

[26] Z. Chen, S. Kiami, A. Gupta, and V. Kumar. Genaug: Retargeting behaviors to unseen situations via generative augmentation. arXiv preprint arXiv:2302.06671, 2023.

[27] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models, 2022. URL https://arxiv.org/abs/2010.02502.

[28] X. Liu, K. Y. Ma, C. Gao, and M. Z. Shou. Diffusion models in robotics: A survey. 2025.

[29] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis, 2021. URL https://arxiv.org/abs/2105.05233.

[30] Y. Du, C. Durkan, R. Strudel, J. B. Tenenbaum, S. Dieleman, R. Fergus, J. Sohl-Dickstein, A. Doucet, and W. Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc, 2023.

[31] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18, pages 234–241. Springer, 2015.

[32] W. Peebles and S. Xie. Scalable diffusion models with transformers, 2023. URL https://arxiv.org/abs/2212.09748.

[33] L. Zhang, A. Rao, and M. Agrawala. Adding conditional control to text-to-image diffusion models, 2023. URL https://arxiv.org/abs/2302.05543.

[34] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020.

[35] A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, and S. Levine. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. arXiv preprint arXiv:1709.10087, 2017.

[36] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. arXiv preprint arXiv:2108.03298, 2021.

[37] Y. Zhu, J. Wong, A. Mandlekar, R. Martín-Martín, A. Joshi, S. Nasiriany, Y. Zhu, and K. Lin. robosuite: A modular simulation framework and benchmark for robot learning. In arXiv preprint arXiv:2009.12293, 2020.

[38] J. Ho and T. Salimans. Classifier-free diffusion guidance, 2022. URL https://arxiv.org/abs/2207.12598.

[39] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. arXiv preprint arXiv:2403.03954, 2024.

[40] H. Zhu, Y. Wang, D. Huang, W. Ye, W. Ouyang, and T. He. Point cloud matters: Rethinking the impact of different observation spaces on robot learning. Advances in Neural Information Processing Systems, 37:77799–77830, 2024.

[41] T.-W. Ke, N. Gkanatsios, and K. Fragkiadaki. 3d diffuser actor: Policy diffusion with 3d scene representations. arXiv preprint arXiv:2402.10885, 2024.

[42] D. Wang, S. Hart, D. Surovik, T. Kelestemur, H. Huang, H. Zhao, M. Yeatman, J. Wang, R. Walters, and R. Platt. Equivariant diffusion policy. arXiv preprint arXiv:2407.01812, 2024.

[43] J. Yang, Z.-a. Cao, C. Deng, R. Antonova, S. Song, and J. Bohg. Equibot: Sim (3)-equivariant diffusion policy for generalizable and data efficient learning. arXiv preprint arXiv:2407.01479, 2024.

[44] S. Haldar and L. Pinto. Point policy: Unifying observations and actions with key points for robot manipulation. arXiv preprint arXiv:2502.20391, 2025.

[45] M. Torne, A. Simeonov, Z. Li, A. Chan, T. Chen, A. Gupta, and P. Agrawal. Reconciling reality through simulation: A real-to-sim-to-real approach for robust manipulation. arXiv preprint arXiv:2403.03949, 2024.

[46] Y. Zhu, A. Joshi, P. Stone, and Y. Zhu. Viola: Imitation learning for vision-based manipulation with object proposal priors. In Conference on Robot Learning, pages 1199–1210. PMLR, 2023.

[47] C. Doersch. Tutorial on variational autoencoders. arXiv preprint arXiv:1606.05908, 2016.

[48] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. Advances in neural information processing systems, 34:8780–8794, 2021.

[49] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017. URL https://arxiv.org/abs/1412.6980.