Robotics: Science and Systems 2026
Sydney, Australia, July 13–17, 2026

RAG-Diff: Adapting Diffusion Policies to Dynamic Constraints with Retrieval-Augmented Guidance

Ruolin Ye[1], Nayoung Ha[1], Shuaixing Chen[1], Qiandao Liu[1], Gavin Chen[1], Shaoyang Stassen[1], Mark Zolotas[2], Jose Barreiros[2], Tapomayukh Bhattacharjee[1]

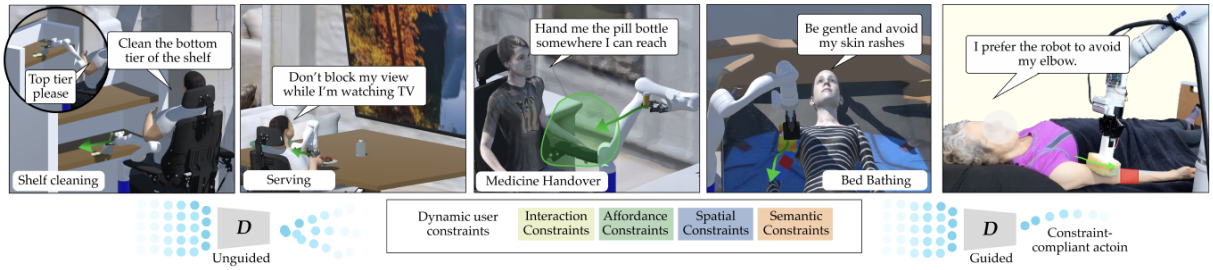


Fig. 1: RAG-Diff (Retrieval-Augmented Guided Diffusion) adapts a *frozen* diffusion policy at runtime, enabling it to satisfy diverse user-specific constraints, including implicit and explicit preferences during execution.

***Abstract*—Robots operating in unstructured environments must satisfy dynamic constraints that can change across tasks and even within a single execution. While diffusion policies can learn multimodal behaviors from demonstrations, adapting a trained policy at runtime to newly encountered or evolving constraints remains an open challenge.**

We propose *RAG-Diff* (Retrieval Augmented Guided Diffusion), a runtime adaptation framework for a frozen transformer diffusion policy that leverages retrieval-augmented memory. RAG-Diff maintains PrefMem, a memory bank that stores vision-language embeddings together with (i) state-action snippets and (ii) constraint annotations. At test-time, RAG-Diff queries PrefMem to retrieve the nearest entry and uses it to steer sampling in two complementary ways. First, I-Atten (In-place Attention recomputation) inserts the retrieved snippet as additional cross-attention memory tokens and performs a classifier-free-guidance-style update, biasing denoising toward preference-consistent motion. Second, a predictive guidance mechanism incorporates the retrieved constraint parameters during diffusion sampling to discourage violations.

To demonstrate the effectiveness of RAG-Diff, we choose physical robot caregiving as a domain with personalized and time-varying constraints. We first benchmark on an adapted PushT environment in simulation with contact-force limits and region-to-avoid constraints. We then evaluated our method on a suite of physical caregiving tasks spanning diverse preference types: (i) interaction and affordance preferences in bed bathing, (ii) ROM-based assistance-level preferences in medicine delivery, (iii) semantic preferences in shelf cleaning, and (iv) trajectory preferences in feeding, in both RCareWorld simulation and with a real robot. We further conducted real-world user studies on a bed-bathing task. Results show that RAG-Diff improves both task success and constraint satisfaction compared to a range of baselines, including unguided diffusion and other guidance- or sampling-based variants. Website: https://emprise.cs.cornell.edu/rag-diff/.

[1]Cornell University. {ry273, nh285, ql342, gc487, ses439, tapomayukh}@cornell.edu, alkdischen@gmail.com
[2]Toyota Research Institute. {mark.zolotas, jose.barreiros}@tri.global

I. Introduction

Robots in unstructured environments must satisfy *dynamic constraints* that evolve both across and within tasks. Beyond achieving a high-level goal, these robots must handle the evolving constraints such as modulating contact forces when transitioning from scrubbing a stain to wiping a fragile glass surface, or instantaneously rerouting to avoid a human arm reaching into the workspace. A capable robot must therefore (i) adapt to these constraints online and (ii) do so with minimal additional training, trial-and-error, or human intervention.

To acquire manipulation skills that are necessary for these tasks, diffusion policies have recently emerged as a strong paradigm for learning multimodal visuomotor behaviors [7, 16, 25, 30]. However, while effective for learning a fixed distribution of behaviors, efficiently adapting these policies at runtime to satisfy dynamic constraints that are not fully specified during training remains a challenge.

Prior approaches for adapting diffusion policies rely on assumptions that limit their use in tasks with varying constraints. Mainstream methods incorporate guidance signals through *fine-tuning* or by *embedding constraints into training*, which makes them slow to respond to new or previously unseen constraints [1, 4, 6, 15, 26]. Another line of work adopts *sample-and-rank*: generating multiple candidate rollouts and selecting the best one according to a metric. While effective as post-hoc selection, these methods cannot exceed the representational ceiling of the underlying policy and become inefficient when rollouts must be repeatedly sampled and evaluated [31]. Finally, *classifier-guided* variants train an additional model (e.g., a classifier or reward predictor) to steer sampling [8, 25]. In settings with evolving constraints, classifier-guided approaches face two practical challenges. First, many constraints may be implicit, context-dependent, or hard to label densely enough to supervise a reliable predictor. Second, without a mechanism to recall past adjustments, they

suffer from redundant computation in recurring situations.

To address these limitations, we propose RAG-Diff (**R**etrieval **A**ugmented **G**uided **Diff**usion) that uses retrieval-augmented memory to adapt diffusion policies at runtime. RAG-Diff freezes a base diffusion policy trained on generic demonstrations, and augments it at inference-time with **PrefMem**, a lightweight memory bank that stores visual contexts paired with action snippets and constraint annotations. At each step, a frozen vision-language encoder maps the recent observation history to an embedding used for nearest-neighbor retrieval, returning (i) a relevant snippet with actions and states and (ii) the associated constraint. We use the retrieved snippets and constraints to guide the diffusion process in two ways. Firstly, we propose **I-Atten** (**I**n-place **Atten**tion recomputation) that injects the retrieved snippet into the diffusion transformer as additional cross-attention memory tokens and perform in-place attention recomputation during denoising. Second, to apply explicit constraints, we feed the retrieved constraint parameters into a predictive guidance mechanism. These two ways of guidance steer the diffusion denoising process together.

While our method is potentially applicable to a broad class of tasks, we showcase its effectiveness in the *physical robot caregiving* domain, which captures key challenges common to tasks with dynamic constraints. In this setup, constraints take the form of user preferences that vary not only across individuals but even for the same person throughout the day as their clinical state changes. For example, a user with upper-limb spasticity may tolerate faster washing and contact force after morning stretching and medication, but require slower motion with lighter contact later in the day when fatigue increases tone and spasms, making rapid movements uncomfortable or unsafe [13]. We consider two practically important classes of user constraints: (i) *interaction constraints*, such as limiting contact force for comfort and safety; and (ii) *affordance constraints*, such as *spatial* restrictions on where the robot may act, e.g., avoiding irritated or rash-prone areas during bed bathing. We capture each constraint with a differentiable value function that measures the magnitude of violation over a candidate action trajectory, and use this value function to steer inference toward constraint-satisfying behaviors. In addition, the value functions naturally supports safety margins and composes across multiple constraints via weighted summation, enabling stable enforcement of simultaneous comfort and spatial requirements.

We evaluate RAG-Diff in both simulation and real-world settings on tasks that capture interaction constraints and affordance constraints in the context of physical robot caregiving. In simulation, we evaluate on an adapted PushT [7, 9] benchmark with contact force and region-to-avoid constraints. It serves as a toy environment representing typical physical caregiving tasks requiring goal completion under user-specific interaction and affordance constraints. We further evaluate our approach across four everyday physical caregiving tasks in RCareWorld [29]: (1) bed bathing, where the robot must respect contact-force limits and avoid sensitive skin regions; (2) medication delivery, where handover locations must adapt to the user's range of motion; (3) shelf cleaning, where the robot prioritizes user-preferred tiers and avoids undesired areas; and (4) feeding, where the robot brings a dish while the user watches TV without occluding their line of sight. We evaluate these task in the realworld with a manikin, and did user study for bed bathing with 11 users, including 2 older adults. The results show that RAG-Diff improves task success while consistently satisfying user preference constraints.

This paper's main contributions are as follows:

- We propose **I-Atten** (**I**n-place **Atten**tion recomputation), a test-time method for transformer-based diffusion policies that injects retrieved snippets as additional cross-attention memory tokens and steers denoising via a classifier-free-guidance-style update, without retraining.
- We propose **PrefMem**, a retrieval-augmented preference memory that stores tuples of visual-context embeddings, action snippets, and user constraints, enabling the robot to retrieve relevant behaviors and recall task constraints across recurring situations.
- We complement retrieval prompting with predictive guidance for structured constraints, allowing direct constraint enforcement during diffusion sampling.
- We evaluate RAG-Diff in simulation, on a real robot, and with human-subject user studies with older adults and younger adults wearing motion-restricting occupational therapy bands to emulate mobility limitations. We find that RAG-Diff consistently outperforms strong baselines in both task success and preference constraint satisfaction.

II. Related Work

A. Guided Diffusion Policies

Prior work on steering diffusion policies typically follows one of the four strategies as shown in Table I.

The first strategy fine-tunes the policy, often with reinforcement learning (RL) [4, 6, 15, 26]. Fine-tuning can adapt a policy to a fixed domain, but it requires a costly training loop. As a result, this strategy struggles to keep up with dynamic constraints and may be unsuitable in safety-critical settings where trial-and-error is too risky. The second strategy embeds constraints during training, commonly through classifier-free (CF) guidance [1]. CF guidance encodes constraints as conditioning tokens so the policy can follow them at inference time, but it generally cannot follow guidance that was not provided during training. The third strategy uses sample-and-rank (SR), which generates many candidate rollouts and selects the best using a metric [31]. SR avoids training and offers flexibility, but it becomes inefficient when rollouts must be repeatedly sampled and evaluated, and it cannot exceed the representational capacity of the underlying policy. The final strategy applies classifier-guided (CG) or gradient-based inference-time guidance [8, 14]. This strategy trains an auxiliary model and uses its gradients to steer sampling, which allows new guidance at test-time. However, evolving constraints expose two limitations of the CG strategy: (i) implicit constraints

TABLE I: Comparison with prior diffusion-policy adaptation methods. Columns indicate whether a method (i) avoids finetuning at test-time, (ii) can incorporate guidance signals not specified during training, (iii) supports multiple simultaneous guidance signals, (iv) reuses computation for recurring (previously seen) conditions, and (v) can be guided via language. We also categorize the primary guidance mechanism as reinforcement learning fine-tuning (RL), sample-and-rank (SR), classifier-free guidance (CF), classifier guidance (CG), or a mixed strategy.

Method	(i) Finetuning-free	(ii) Untrained Guidance	(iii) Multiple Guidance	(iv) Seen-Condition Efficiency	(v) Guiding through Language	Guidance Type
DiWA [4]	✗	✗	✗	✗	✗	RL
Venkatraman et al. [26]	✗	✗	✗	✗	✗	RL
AdaptDiffuser [15]	✗	✗	✗	✗	✗	RL
FDPP [6]	✗	✗	✗	✗	✗	RL
D-MPC [31]	✓	✓	✓	✗	✗	SR
Ajay et al. [1]	✓	✗	✓	✗	✗	CF
Diffuser [14]	✓	✓	✓	✗	✗	CG
DynGuide [8]	✓	✓	✓	✗	✗	CG
ITPS [27]	✓	✓	✓	✗	✗	Mixed
Ours	✓	✓	✓	✓	✓	Mixed

are hard to supervise reliably, and (ii) the method exhibits inefficiency by recomputing guidance from scratch even in recurring scenarios. In comparison, our RAG-Diff method adapts at inference-time without fine-tuning or constraint-specific training, and improves efficiency in recurring scenarios by reusing past adjustments rather than recomputing guidance from scratch. ITPS [27] provides a thorough and insightful benchmark of inference-time steering strategies for frozen diffusion policies under real-time user inputs such as points, sketches, and physical corrections. They study six sampling strategies, including post-hoc ranking (SR), biased initialization, guided diffusion (CG), and stochastic sampling. Their analysis reveals a fundamental alignment–constraint satisfaction trade-off: aggressive steering improves alignment but pushes samples off the data manifold, while stochastic sampling achieves the best trade-off by sampling from the product rather than the sum of distributions. Our value guidance falls into the guided-diffusion category of their taxonomy. We build on the value guidance framework by extending steering to retrieval-augmented preferences and our complementary I-Atten mechanism, allowing our method to adapt to not only the explicit constraints, but also the implicit preferences of the users.

Beyond explicit inference-time guidance, previous works have also applied diffusion policies to contact-rich manipulation by modeling compliance and reactive feedback. ACP [12] diffuses over both reference actions and target stiffness to learn spatially and temporally varying compliance, while RDP [28] and ImplicitRDP [5] use slow-fast designs to combine action chunking with high-frequency tactile-reactive corrections. In contrast, we adapt a frozen diffusion policy via predictive guidance to enforce interaction constraints (e.g., force thresholds) during sampling, without explicitly learning a compliant controller or retraining.

B. Retrieval-augmented policies

Retrieval-augmented policies store an explicit memory of past experience. At test-time, they retrieve context-matched sub-trajectories or latent plans from demonstrations or prior rollouts to improve decision making. Most prior work adopts the assumption that similar visual observations imply similar action trajectories, and therefore performs retrieval in an embedding space computed from visual encoders [20, 22]. Such retrieval has been used to augment non-diffusion policies by treating retrieved content as a static input, e.g., retrieving sub-trajectories or skills to condition policy learning and execution [20, 22]. We follow this paradigm and use VLM-based visual embeddings to retrieve nearest-neighbor snippets relevant to the current scene. Among diffusion-policy methods, the closest to our use of retrieval is RAGDP [23]. However, RAGDP uses retrieval mainly to speed up inference by reducing denoising steps via nearest-neighbor expert actions, while our approach guides the denoising process with the retrieved samples to shape the generated action trajectories.

C. Preference Learning for Physical HRI

Preference learning for physical human-robot interaction (pHRI) spans multiple dimensions, including preferences related to physical contact [17, 18], as well as motion and trajectory execution [2, 21]. Prior work on personalization in pHRI has largely focused on task-specific assistive scenarios, such as bathing [17, 32], dressing [10], or feeding [3]. While effective within their respective domains, such methods limit generalization to other pHRI tasks.

More recent work has shifted toward task-agonistic formulations. However, existing approaches often lack long-term adaptation [18], or encoded within constrained planner frameworks [24], limiting their expressiveness compared to policy-level adaptation.

To address these limitations, our method targets learned diffusion policies and focuses on *inference-time* adaptation: it injects retrieved, preference-labeled experience directly into the denoiser's internal computation to shape action generation diffusion action denoising.

III. Problem Formulation

We study test-time steering of a *frozen* action diffusion policy under *time-varying user preferences*. At step t, the policy conditions on a recent observation window $o_{t-T_o+1:t}$ of length T_o and produces an action chunk of length T_a. A conditional diffusion sampler maintains a sequence of noisy actions of $A_t^k \in \mathbb{R}^{T_a \times D_a}$ over K denoising steps $k \in \{K, \dots, 0\}$, starting from $A_t^K \sim \mathcal{N}(0, \mathbb{I})$ and iteratively denoising until the final chunk A_t^0. A vanilla diffusion policy thus induces the conditional distribution

$$A_t^0 \sim D_\theta(\cdot \mid o_{t-T_o+1:t}). \tag{1}$$

We represent the user preference at time t as constraint parameters p_t (e.g., a force threshold F_{th}, an affordance SDF d_{afford}, a goal or a language request g). Our objective is to complete the task while minimizing preference violations by steering sampling at test-time without updating the base parameters θ. We propose RAG-Diff towards this end.

IV. Method

RAG-Diff has two stages: we first train a base action diffusion policy from offline demonstrations (Sec. IV-A), and then steer the *frozen* policy at test-time to satisfy varying user preferences (Sec. IV-B). We initialize a preference memory pool from the same demonstrations, which provides retrievable state-action snippets to guide *implicit* preferences via ***I-Atten*** (Sec. IV-B0a) and the corresponding constraint parameters to guide *explicit* preferences via value guidance (Sec. IV-B0b).

A. Training a base action diffusion policy

We train a *transformer-based* diffusion policy to model a distribution over action chunks conditioned on recent observations, following Diffusion Policy [7]. Given an observation history $o_{t-T_o+1:t}$ at timestep t consisting of RGB images and low-dimensional states (e.g., robot joint states and human pose), we embed the current noisy action chunk into *action embeddings* and embed the observation history into *observation embeddings*. Following Diffusion Policy [7], we extract visual observation embeddings with a ResNet and concatenate them with embedded low-dimensional states to form a cross-attention context $C \in \mathbb{R}^{T_c \times d}$, where T_c is the number of context tokens and d is the transformer hidden dimension. We generate an action chunk by running a denoising process over $A_t^k \in \mathbb{R}^{T_a \times D_a}$. Unless otherwise specified, the action A denotes the robot end-effector position trajectory. At each denoising step k, the transformer denoiser takes the current noisy action tokens A_t^k and cross-attends to C to predict a noise residual

$$\epsilon_{\text{base}} \;=\; \epsilon_\theta(A_t^k, k \mid C). \tag{2}$$

B. Guiding the action diffusion process at test time

We guide the diffusion process in two complementary ways. (1) ***I-Atten*** targets preferences that are *implicit*, i.e., best conveyed by example rather than an explicit parameter. It steers sampling by injecting similar state-action snippets as additional cross-attention context, encouraging the policy to reuse appropriate behavior modes. However, retrieval-based prompting can be approximate and does not explicitly enforce constraints. We therefore also apply (2) value guidance, which directly biases denoising using gradients of differentiable constraint objectives to satisfy user-specified preferences.

a) I-Atten: In-place attention recomputation: I-Atten steers sampling by injecting a state-action snippet as additional cross-attention context. Given a prompt snippet with paired observations and actions, we encode its observations using the same encoders as the base policy. Specifically, we extract ResNet features from the RGB frames and concatenate them with embedded low-dimensional states. We embed the paired actions in the same action-token space. This yields prompt tokens $P \in \mathbb{R}^{T_p \times d}$. We then concatenate it with the original context C:

$$C^+ = [\, C;\, P \,] \in \mathbb{R}^{(T_c+T_p)\times d}. \tag{3}$$

In our implementation, we obtain the prompt snippet by retrieving a relevant segment from the preference memory pool $\mathcal{B}$ (Sec. IV-C). We run the denoiser twice on the same noisy chunk, differing only in the cross-attention context:

$$\epsilon_{\text{base}} = \epsilon_\theta(A_t^k, k \mid C), \qquad \epsilon_{\text{prompt}} = \epsilon_\theta(A_t^k, k \mid C^+), \tag{4}$$

and mix the two predictions with a classifier-free-guidance(CFG)-style update,

$$\epsilon_{\text{attn}} = \epsilon_{\text{base}} + w_{\text{attn}}(k)\big(\epsilon_{\text{prompt}} - \epsilon_{\text{base}}\big), \tag{5}$$

where $w_{\text{attn}}(k) = s_{\text{attn}}\alpha_{\text{attn}}(k)$ sets the guidance weight, with s_{attn} controlling the overall strength and $\alpha_{\text{attn}}(k)$ scheduling that strength across denoising steps.

b) Constraints as value guidance: I-Atten provides *implicit* guidance by biasing sampling toward retrieved state-action snippets, but it does not explicitly guide the policy. To handle *explicit* user preferences, we impose constraints via a differentiable value function defined on the sampler's current estimate of the clean action trajectory. At step t and denoising step k, we recover the predicted clean action chunk $\hat{A}_t^0$ from the current noisy iterate A_t^k and the denoiser output ϵ_{base} via the standard Tweedie relation

$$\hat{A}_t^0 \;=\; \frac{A_t^k - \sqrt{1-\bar{\alpha}_k}\,\epsilon_{\text{base}}}{\sqrt{\bar{\alpha}_k}}, \tag{6}$$

and define $J(\hat{A}_t^0)$ to measure preference violations over the horizon. We then apply gradient-based guidance by perturbing the noise prediction:

$$\epsilon_{\text{value}} \;=\; \epsilon_{\text{base}} \;+\; w_{\text{value}}(k)\,\Delta\epsilon(\hat{A}_t^0), \tag{7}$$

$$\Delta\epsilon(\hat{A}_t^0) \;=\; \frac{\sqrt{\bar{\alpha}_k}}{\sqrt{1-\bar{\alpha}_k}}\,\nabla_{\hat{A}_t^0} J(\hat{A}_t^0), \tag{8}$$

where $w_{\text{value}}(k) = s_{\text{value}}\alpha_{\text{value}}(k)$ scales the guidance by a strength s_{value} and a schedule $\alpha_{\text{value}}(k)$. In RAG-Diff, we consider the following classes of preference constraints. More broadly, the value guidance formulation is flexible and compositional, and can readily incorporate additional preference types by defining the corresponding differentiable value functions [14].

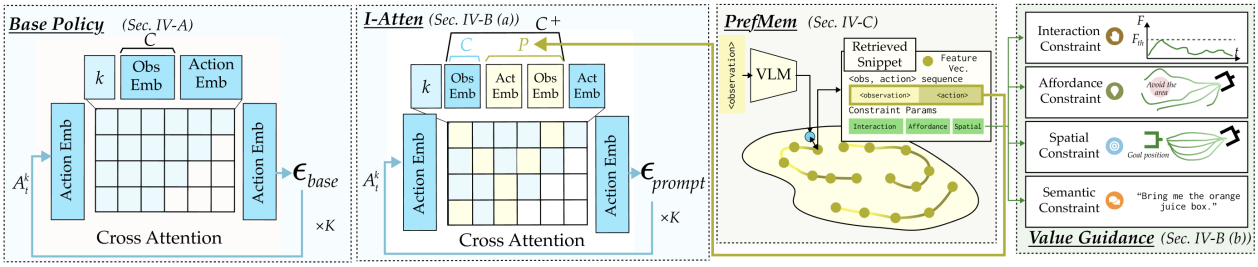


Fig. 2: **Method overview.** RAG-Diff adapts a *frozen* diffusion policy at test time to satisfy user preferences. At each timestep, we embed the current observation history with a VLM encoder and retrieve relevant snippetsfrom a memory pool. We then guide denoising in two complementary ways: (i) *in-place attention recomputation* injects the retrieved snippet as additional cross-attention memory tokens and steers sampling via a classifier-free guidance (CFG)-style update, and (ii) *value guidance* enforces interaction and affordance constraints by injecting gradients of a differentiable constraint during sampling.

Interaction (force) constraint. Safe physical interaction is central to physical robot caregiving, and contact force provides a direct, widely used indicator of comfort and safety, so we enforce a user-specific force threshold.

We constrain the predicted force magnitudes to remain under a comfort threshold F_{th}. We evaluate the constraint on the predicted clean action chunk $\hat{A}_t^0 = \{\hat{a}_\tau\}_{\tau=0}^{T_a-1}$. Using a differentiable force predictor g_ϕ (MLP) on state–action pairs $\hat{f}_\tau = g_\phi(s_\tau, \hat{a}_\tau)$, we define a soft-hinge value function

$$J_{\text{force}}(\hat{A}_t^0) = \sum_{\tau=0}^{T_a-1} \Big[\max\big(0,\ \hat{f}_\tau - F_{\text{th}}\big)\Big]^2. \tag{9}$$

Minimizing J_{force} penalizes threshold violations, steering sampling toward trajectories that maintain comfortable contact.

Affordance (region to avoid) constraint. Caregiving often involves sensitive areas such as wounds or skin rashes that should be avoided. Modeling sensitive areas as an avoid zone provides a smooth distance-to-unsafe gradient that steers the end-effector away without derailing the task.

We encode spatial preferences with a differentiable signed-distance function (SDF) $d_{\text{afford}}(\cdot)$, where $d(u) > 0$ is outside the region and $d(u) < 0$ is inside. With margin m, we define the value function

$$J_{\text{afford}}(\hat{A}_t^0) = \sum_{\tau=0}^{T_a-1} \Big[\max\big(0,\ m - d_{\text{afford}}(\hat{a}_\tau)\big)\Big]^2. \tag{10}$$

Minimizing J_{afford} penalizes trajectories that approach or enter the avoid region, steering sampling to maintain a safe distance from that region across the horizon.

Spatial (goal position) constraint. We model spatial preferences as a goal end-effector position because many user intentions reduce to "place/reach here/there," and a distance-to-goal objective gives a smooth, differentiable signal that can steer denoising directly.

Given a target position $g \in \mathbb{R}^3$, we define

$$J_{\text{goal}}(\hat{A}_t^0, g) = \frac{1}{T_a} \sum_{\tau=0}^{T_a-1} \big\|\hat{a}_\tau - g\big\|_2^2. \tag{11}$$

Minimizing J_{goal} penalizes the average squared distance between the predicted end-effector positions and the target goal, encouraging the trajectory to move toward the goal throughout the horizon.

Semantic (language-grounded) constraint. Semantic requests (e.g., "bring the orange juice") do not directly specify geometry. We ground them by using a VLM to retrieve relevant snippets from a memory bank, and then enforce it with value guidances (Detailed in Sec. V-E).

We combine constraints with a weighted sum,

$$J(\hat{A}_t^0) = \lambda_F\, J_{\text{force}}(\hat{A}_t^0) + \lambda_A\, J_{\text{afford}}(\hat{A}_t^0) + \lambda_S\, J_{\text{goal}}(\hat{A}_t^0), \tag{12}$$

and inject $\nabla_{\hat{A}_t^0} J(\hat{A}_t^0)$ during denoising to steer.

c) Summary: We summarize one reverse step of RAG-Diff as follows:

$$\begin{aligned}\epsilon_{\text{total}} = &\underbrace{\epsilon_\theta(A_t^k, k \mid C)}_{\text{Base: Sec. IV-A}} \\ &+ \underbrace{w_{\text{attn}}(k)\Big(\epsilon_\theta(A_t^k, k \mid C^+) - \epsilon_\theta(A_t^k, k \mid C)\Big)}_{\text{I-Atten: Sec. IV-B0a}} \\ &+ \underbrace{w_{\text{value}}(k)\frac{\sqrt{\bar{\alpha}_k}}{\sqrt{1-\bar{\alpha}_k}}\nabla_{\hat{A}_t^0} J(\hat{A}_t^0)}_{\text{Value Guidance: Sec. IV-B0b}}.\end{aligned} \tag{13}$$

C. PrefMem: Preference Memory and Retrieval

Both guidance modules introduced in the previous section rely on preference-relevant context. *I-Atten* requires a state-action snippet to use as a prompt, and *value guidance* requires the corresponding constraint parameters (e.g., a force threshold F_{th}, an avoid region d_{afford}, or a goal g). We obtain both from a shared memory pool, *PrefMem*, initialized from the same offline demonstrations used to train the diffusion policy. Each entry stores the observation (RGB frames and low-dimensional states, such as robot joint positions and human pose) together with the associated preference annotation.

a) State encoding: At each step t, we form a retrieval query from the same visual observation stream used by the base diffusion policy. Let $o_{t-T_o+1:t}$ denote the observation history, and let $\{I_{t-T_o+1}, \ldots, I_t\}$ be the T_o RGB frames in

this history. We compute a per-frame latent vector using a frozen VLM f_{vlm},

$$z_\tau = f_{\text{vlm}}(I_\tau), \quad \tau \in \{t - T_o + 1, \ldots, t\},$$

and aggregate them by average pooling to obtain a single query vector,

$$\tilde{q}_t = \frac{1}{T_o} \sum_{\tau=t-T_o+1}^{t} z_\tau, \qquad q_t = \frac{\tilde{q}_t}{\|\tilde{q}_t\|_2}.$$

We use q_t for nearest-neighbor retrieval in the preference memory, motivated by the assumption that visually similar observation histories often induce similar action snippets and similar preferences. Retrieved items provide both (i) a state-action snippet and (ii) the associated preference annotation p_t used to parameterize downstream guidance.

b) Memory bank: We construct a retrieval bank $\mathcal{B}$ from the same offline demonstrations used to train the base diffusion policy. Specifically, we segment each demonstration trajectory into overlapping windows and store tuples

$$b_i = \Big(q_i,\ \mathbf{I}_i,\ \mathbf{s}_i,\ \mathbf{a}_i,\ p_i \Big) \in \mathcal{B},$$

where $\mathbf{I}_i = \{I_{i-T_o+1}, \ldots, I_i\}$ is a length-T_o sequence of RGB frames, $\mathbf{s}_i = \{s_{i-T_o+1}, \ldots, s_i\}$ is the low-dimensional state history, $q_i \in \mathbb{R}^d$ is the normalized VLM embedding of the observation window, $\mathbf{a}_i = a_{i:i+T_a-1}$ is the corresponding length-T_a action snippet, and p_i denotes the associated user-preference annotation for that snippet (e.g., a force threshold, a region to avoid, or a semantic preference). The memory bank scales efficiently: retrieval latency remains nearly constant as memory size grows, while memory and storage increase linearly with the number of stored snippets, supporting scalable real-time retrieval (see Appendix for details).

c) Top-N retrieval: At test-time, given the query vector q_t, we retrieve the top N entries in $\mathcal{B}$ by cosine similarity. We use the retrieved snippets as prompt for in-place attention recomputation, and use the corresponding preference annotations to parameterize the value guidances during sampling with the frozen base policy.

d) Incorporating new user preferences at test-time: When the user provides a new preference p_t^{new} at time t, we first determine whether it can be associated with an existing memory entry by retrieving the nearest neighbors under cosine similarity, and gets the cosine similarity scores s_t.

If $s_t \geq \delta$ for a similarity threshold δ, we treat the preference as applicable to a previously seen context and update only the stored preference with the new one. Otherwise, if $s_t < \delta$, we regard the preference as out-of-distribution with respect to the current bank and extend memory using the newly observed interaction. Here, we assume the user's preference will not change within the time window of this snippet. We set δ empirically by inspecting the distribution of cosine similarities in the VLM latent space and choosing δ at the gap between high-similarity (in-cluster) matches and low-similarity (out-of-cluster) matches.

V. Experiments

We evaluate RAG-Diff across three complementary settings: (1) a suite of simulated tasks with various constraints to benchmark performance and systematically compare against baselines; (2) the same tasks executed on a real robot, validating real-world feasibility; and (3) user studies with human subjects, measuring preference satisfaction during interaction. We use RCareWorld [29] for all simulation experiments, and a Kinova Gen3 arm with a Realsense D435 camera mounted on the wrist for all real-world evaluations.

Our evaluation aims to answer the following questions:

- **Q1:** Under fixed preferences, is RAG-Diff improving task success while reducing constraint violations more effectively than the baselines?
- **Q2:** When preferences change during execution, does RAG-Diff adaapt to new constraints?

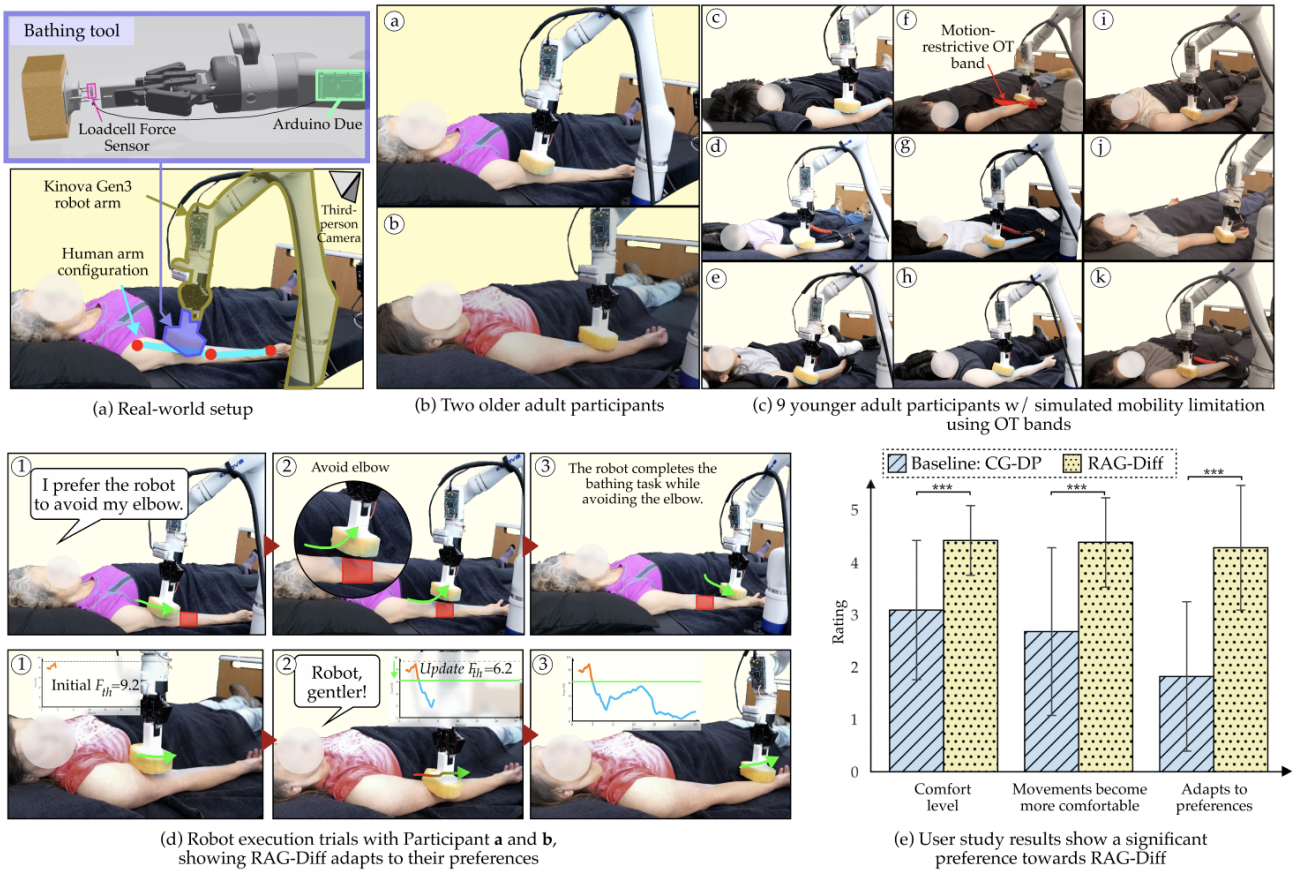
A. Tasks and environments

We evaluate RAG-Diff in both simulation and real-world settings on a diverse set of manipulation and physical caregiving tasks designed to span various preference types, including *interaction constraints*, *affordance constraints*, and *spatial constraints*. We evaluate each task in simulation and on real robot (except for Push-T) using a Kinova Gen3 arm interacting with a manikin. We show the tasks in Fig. 1.

- **Push-T (simulation only)**. A toy environment task where the robot must push a T-block to a goal pose while respecting a force limit and avoiding a specified region. We use the same success criteria as Diffusion Policy [7].
- **Bed Bathing**. The robot must wipe a target region, avoid sensitive areas, and keep contact forces below a comfort threshold of 10N. We define success as $> 75\%$ paint removed for the arm based on the result from a SOTA bathing system [11].
- **Medicine Handover**. The robot presents the bottle at an accessible handover position. We define success as bottle delivered into the care recipient's reachable ROM region.
- **Serving**. The robot brings a dish and places it without occluding the care recipient's view. We define success as dished placed in a place not occluding the care recipient's view.
- **Shelf Cleaning**. The robot cleans the user-preferred tier (upper vs. lower), reflecting a spatial preference over where to act. We define success as 80% of the stain removed on the shelf. The stain only appears on the tier that the user wants the robot to clean which means the robot will automatically fail if it goes to a wrong tier.

B. Baselines and ablations

We compare against the following baselines and ablations. For baselines, we include strong SOTA baselines including **DP** to quantify the base policy without adaptation, **CG-DP** as a strong gradient-based guidance baseline, and **SR** as a strong sampling-based alternative that uses post-hoc selection instead of guidance.



(a) Real-world setup

(b) Two older adult participants

(c) 9 younger adult participants w/ simulated mobility limitation using OT bands

(d) Robot execution trials with Participant **a** and **b**, showing RAG-Diff adapts to their preferences

(e) User study results show a significant preference towards RAG-Diff

Fig. 3: We conducted a real-world user study to evaluate the effectiveness and adaptability of RAG-Diff during robot-assisted bed bathing. Participants interacted with the robot and provided preference updates (e.g., gentler contact and avoiding sensitive regions) during execution. The error bars represent the standard deviation.

- **Diffusion Policy (DP) [7]**: The base diffusion policy without test-time guidance.
- **Classifier-Guided DP (CG-DP)**: The DP policy guided with classifier guidance.
- **Sample-and-Rank (SR)**: Sample multiple action chunks from DP and select the best one via post-hoc scoring.

For ablations, we include **I-Atten DP** and **V-DP** as ablations to isolate the effects of retrieved-context injection versus explicit constraint optimization.

- **I-Atten only (I-Atten DP)**: DP with only I-Atten guidance.
- **Value only (V-DP)**: DP with only value-guidance.
- **I-Atten + Value (RAG-Diff)**: Our full method combining I-Atten and value guidance on DP.

C. Evaluation with constant preferences (Q1)

We first evaluate RAG-Diff under constant user preferences, where the constraints remain fixed throughout evaluation. We show the results in Table II. The results suggest our method improves both task completion and preference compliance under fixed constraints, and that these gains are consistent across simulation and the real world. Overall, **RAG-Diff** performs best across tasks. It achieves the strongest success while simultaneously reducing interaction and affordance violations, indicating that combining retrieved-context injection with value/constraint guidance yields complementary benefits. In contrast, **CG-DP** consistently improves constraint satisfaction but reduces task success. **SR** provides modest improvements by post-hoc selection, yet it is generally less effective than guidance-based methods, highlighting the benefit of steering the denoising process directly rather than selecting from unguided samples. **I-Atten DP** typically improves success relative to the unguided policy, but does not reliably reduce violations, suggesting that attention-based prompting primarily selects favorable trajectory modes without explicitly optimizing constraint satisfaction. **V-DP** offers a more balanced trade-off, improving constraint satisfaction with less impact on success than CG-DP, but remains worse than the full method.

D. User study: Evaluating with dynamic preferences (Q2)

We conducted a user study evaluating RAG-Diff in the context of a robot-assisted bed bathing task. Throughout the study, we aimed to evaluate the performance of RAG-Diff and see how it adapts to user preferences during real world interactions. We evaluated RAG-Diff across 11 participants, including 9 younger adults without mobility limitations (mean age 22.4 years), and two older adults who are 63 and 64 years old respectively. We use a motion-restricting occupational therapy band to simulate mobility limitations in our younger

Simulation										Real-world experiments with a manikin							
	Push-T			Bed Bathing			Med. Hand.	Serv.	Shelf Clean.		Bed Bathing			Med. Hand.	Serv.	Shelf Clean.	
Method	Succ↑	Force↓	Afford↓	Succ↑	Force↓	Afford↓	Succ↑	Succ↑	Succ↑	**Method**	Succ↑	Force↓	Reg↓	Succ↑	Succ↑	Succ↑	
DP	0.80	0.25±0.04	0.42±0.12	0.62	0.32±0.14	0.48±0.18	0.61	0.61	0.43	DP	5/10	0.32±0.14	0.48±0.18	6/10	6/10	4/10	
CG-DP	0.74	0.11±0.03	**0.14±0.06**	0.54	0.18±0.12	0.22±0.07	0.52	0.72	0.48	CG-DP	5/10	0.18±0.12	0.22±0.07	5/10	**8/10**	5/10	
SR	0.82	0.18±0.04	0.30±0.10	0.50	0.22±0.04	0.29±0.17	0.60	0.62	0.51	SR	5/10	0.22±0.04	0.29±0.17	6/10	6/10	5/10	
I-Atten	0.83	0.24±0.03	0.38±0.07	0.70	0.34±0.13	0.46±0.14	0.61	0.70	0.59	I-Atten	6/10	0.34±0.13	0.46±0.14	7/10	6/10	6/10	
V-DP	0.82	0.16±0.03	0.24±0.09	0.66	0.18±0.07	0.22±0.07	0.62	0.69	0.72	V-DP	5/10	0.18±0.07	0.22±0.07	6/10	7/10	6/10	
RAG-Diff	**0.86**	**0.09±0.02**	0.17±0.04	**0.74**	**0.17±0.11**	**0.17±0.08**	**0.81**	**0.83**	**0.84**	**RAG-Diff**	**7/10**	**0.17±0.11**	**0.17±0.08**	**8/10**	**8/10**	**7/10**	

TABLE II: Simulation results in RCareWorld [29] and real-world results with a manikin.

adult participants. Among the participants, 4 are male and 7 are female. Each session lasts 40–60 minutes, with individual wiping interactions taking 1–2 minutes. Participants complete one trial and three evaluation runs, each comparing two methods in **randomized** order. The session duration exceeds the total interaction time due to preparation and a calibration step to initialize user-specific force preferences. The multi-round design enables iterative preference refinement, where users are encouraged to update their preferences online during a trial, though preferences are given at discrete times. The study protocol was reviewed and approved by the Institutional Review Board (IRB) of Cornell University with the protocol number IRB0146211.

1) Study Procedure: Before the study, we assessed the participants' nervousness levels regarding physical assistance from robots in activities of daily living (ADLs) and instrumental activities of daily living (IADLs). Participants rated their levels of nervousness on a likert scale of 1 (not nervous) to 5 (very nervous) and the average score was 2.09.

Before the actual trials, we calibrate the force preference of the participants. Participants lay on the bed with their arm resting naturally at their side while the robot sequentially positioned the bathing tool above the upper arm and forearm to apply gentle downward pressure and gradually increased it. We asked the participants to indicate when the contact began to feel uncomfortable, at which point we immediately stopped the robot. We record the contact force at this moment as the participant's personalized interaction constraint. We then familiarized participants with the system by explaining the setup and procedure, and showing example videos of the robot's motions.

Throughout each user study we perform two trials of interaction, one using RAG-Diff and the other using CG-DP as our baseline method since it is the strongest baseline method. Since the classifier can not adapt during test time, we train the classifier using 10N as an initial threshold that we got from a pilot study with 3 participants. We use an area around the arm pit as the initialization for the affordance classifier following a study in PrioriTouch [18] that suggests as a population average, people would like to avoid this area. At the start of each trial, we applied washable paint along the participant's arm. The robot then attempted to remove the paint by executing a sequence of bathing motions. Each trial consisted of three interaction rounds with both methods. During the trial, we asked participants to verbally prompt the robot to adjust its behavior whenever they want. For example, by requesting a gentler touch or increased pressure. We maintain the memory pool across the three trials.

2) Results: We evaluated the methods across three categories: comfort level of contact, whether the movements became more comfortable over time, and whether the robot was able to adapt to the participant's preferences. Participants rated each category on a 5-point likert scale from 1 (strongly disagree) to 5 (strongly agree).

Overall, RAG-Diff received higher scores than the CG-DP baseline across all three quantitative measures (Fig. 3). Participants rated contact comfort at 3.09 ± 1.33 for the CG-DP baseline, compared to 4.42 ± 0.66 for RAG-Diff. For perceived improvement in comfort over time, RAG-Diff achieved 4.38 ± 0.85, outperforming the CG-DP baseline at 2.68 ± 1.60. Participants also rated preference adaptation higher for RAG-Diff (4.29 ± 1.19) than for the CG-DP baseline (1.82 ± 1.42). A Mann–Whitney U test [19] indicated that RAG-Diff significantly outperformed the CG-DP baseline on all three metrics ($p < 0.005$).

Across all trials, participants preferred RAG-Diff in 81.81% of cases. In trials where preferences changed during the interaction, 100% of participants preferred RAG-Diff, suggesting that the method adapts effectively to preference changes. Participants who favored RAG-Diff over the CG-DP baseline attributed their preference to the robot applying gentler force when requested and reliably avoiding regions they asked the robot to avoid. The users, especially the older adult group praised our robot with *"Old ladies have sensitive skin. I appreciate the robot following my preferences and being gentle with me."* and *"The robot really feels like it's listening to me and responding to what I need"*.

After the study, we asked participants to rate how nervous they felt about a robot physically assisting them. The average rating was 1.66/5 (1 = not nervous, 5 = very nervous).

E. Guidance through language

We also evaluate whether RAG-Diff can incorporate natural-language preferences at test time. In a realworld drink fetching task, we place two drinks, a Coke can and a juice box, on a table and provide a language instruction specifying which drink the user prefers (e.g., *"Bring me the orange juice box."*). We leverage text-to-image retrieval in VLM embedding space, which enables language-conditioned retrieval not supported by baselines. A trial is successful if the robot grasps the correct

object. RAG-Diff succeeds in all trials (5/5) with distinctive language description. In comparison, none of the baselines are able to do guidance through language. While robust for distinct objects/clear language descriptions, it degrades under visual similarity or ambiguous language (e.g., "robot arm with a can" yields the same similarity to both orange juice box and coke can images) due to the VLM limitation.

VI. Discussion and Limitations

We observe failure cases due to misleading retrieval, especially in visually similar but semantically different contexts. We acknowledge this as a limitation of our work. Incorrect retrieval leads to two main failure modes: (1) incorrect preference labels, causing optimization toward mismatched targets and resulting in preference violations, which can potentially be fixed using a correct label during the interaction, and (2) dissimilar retrieved states, leading to noisy action rollouts and degraded performance. Future work could potentially improve this by integrating robot states and other relevant information into the retrieval pipeline.

Another limitation of our method is that performance is sensitive to guidance hyperparameters, especially the guidance scale and its schedule across denoising steps. Empirically, in-place attention recomputation benefits from stronger early guidance, while analytical penalties are most effective later, likely because attention selects a high-level trajectory mode when samples are still noisy, whereas value guidance refines an already-formed trajectory when violations are well-defined. Also, attention prompting assumes retrieved snippets are in-distribution for the frozen base policy; out-of-distribution retrieval (e.g., mismatched dynamics or visual context) can inject misleading signals and degrade performance. Incorporating retrieval OOD detection is an important direction. Third, the method depends on the retrieval encoder quality: weak VLM embeddings can return irrelevant neighbors, and extending retrieval with additional modalities (e.g., force) can be domain-specific and may require extra sensing and representation tuning. Fourth, our assumption that visually similar contexts imply similar actions and preferences may fail when preferences depend on latent factors not observable from RGB. Finally, our user preference constraints are enforced as soft guidance rather than hard safety guarantees; real-world deployment should incorporate hard safety guardrails as well.

Acknowledgements

This work was partly funded by National Science Foundation IIS #2132846, and CAREER #2238792. The Toyota Research Institute also partially supported this work. This article solely reflects the opinions and conclusions of its authors and not TRI or any other Toyota entity. This research was also funded, in part, by the Advanced Research Projects Agency for Health (ARPA-H) Agreement No. 140D042590012. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government.

References

[1] Anurag Ajay, Yilun Du, Abhi Gupta, Joshua B. Tenenbaum, Tommi S. Jaakkola, and Pulkit Agrawal. Is conditional generative modeling all you need for decision making? In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=sP1fo2K9DFG.

[2] Armin Avaei, Linda van der Spaa, Luka Peternel, and Jens Kober. An incremental inverse reinforcement learning approach for motion planning with separated path and velocity preferences. *Robotics*, 12(2):61, April 2023. ISSN 2218-6581. doi: 10.3390/robotics12020061. URL http://dx.doi.org/10.3390/robotics12020061.

[3] Gerard Canal, Guillem Alenyà, and Carme Torras. Personalization framework for adaptive robotic feeding assistance. In Arvin Agah, John-John Cabibihan, Ayanna M. Howard, Miguel A. Salichs, and Hongsheng He, editors, *Social Robotics*, pages 22–31, Cham, 2016. Springer International Publishing. ISBN 978-3-319-47437-3.

[4] Akshay L Chandra, Iman Nematollahi, Chenguang Huang, Tim Welschehold, Wolfram Burgard, and Abhinav Valada. Diwa: Diffusion policy adaptation with world models. *Conference on Robot Learning (CoRL)*, 2025.

[5] Wendi Chen, Han Xue, Yi Wang, Fangyuan Zhou, Jun Lv, Yang Jin, Shirun Tang, Chuan Wen, and Cewu Lu. Implicitrdp: An end-to-end visual-force diffusion policy with structural slow-fast learning. *arXiv preprint arXiv:2512.10946*, 2025.

[6] Yuxin Chen, Devesh K. Jha, Masayoshi Tomizuka, and Diego Romeres. Adapting diffusion policies to human preferences via reward-guided fine-tuning. In *1st Workshop on Safely Leveraging Vision-Language Foundation Models in Robotics: Challenges and Opportunities*, 2025. URL https://openreview.net/forum?id=s3qcfTEsYV.

[7] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[8] Maximilian Du and Shuran Song. Dynaguide: Steering diffusion policies with active dynamic guidance. In *Proceedings of the 39th Conference on Neural Information Processing Systems (NeurIPS)*, 2025.

[9] Pete Florence, Corey Lynch, Andy Zeng, Oscar Ramirez, Ayzaan Wahid, Laura Downs, Adrian Wong, Johnny Lee, Igor Mordatch, and Jonathan Tompson. Implicit behavioral cloning. *Conference on Robot Learning (CoRL)*, November 2021.

[10] Yixing Gao, Hyung Jin Chang, and Yiannis Demiris. User modelling for personalised dressing assistance by humanoid robots. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1840–1845, 2015. doi: 10.1109/IROS.2015.7353617.

[11] Yijun Gu and Yiannis Demiris. Learning bimanual manipulation policies for bathing bed-bound people. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8936–8943, 2024. doi: 10.1109/IROS58592.2024.10801478.

[12] Yifan Hou, Zeyi Liu, Cheng Chi, Eric Cousineau, Naveen Kuppuswamy, Siyuan Feng, Benjamin Burchfiel, and Shuran Song. Adaptive compliance policy: Learning approximate compliance for diffusion guided control. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4829–4836, 2025. doi: 10.1109/ICRA55743.2025.11128452.

[13] Carrie L. Hugos and Michelle H. Cameron. Assessment and measurement of spasticity in ms: State of the evidence. *Current Neurology and Neuroscience Reports*, 19 (10):79, August 2019. doi: 10.1007/s11910-019-0991-2.

[14] Michael Janner, Yilun Du, Joshua B. Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *International Conference on Machine Learning*, 2022.

[15] Zhixuan Liang, Yao Mu, Mingyu Ding, Fei Ni, Masayoshi Tomizuka, and Ping Luo. Adaptdiffuser: Diffusion models as adaptive self-evolving planners. In *International Conference on Machine Learning*, 2023.

[16] Xiao Ma, Sumit Patidar, Iain Haughton, and Stephen James. Hierarchical diffusion policy for kinematics-aware multi-task robotic manipulation. *CVPR*, 2024.

[17] Rishabh Madan, Skyler Valdez, David Kim, Sujie Fang, Luoyan Zhong, Diego T. Virtue, and Tapomayukh Bhattacharjee. Rabbit: A robot-assisted bed bathing system with multimodal perception and integrated compliance. In *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '24, page 472–481, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400703225. doi: 10.1145/3610977.3634989. URL https://doi.org/10.1145/3610977.3634989.

[18] Rishabh Madan, Jiawei Lin, Mahika Goel, Amber Li, Angchen Xie, Xiaoyu Liang, Marcus Lee, Justin Guo, Pranav N. Thakkar, Rohan Banerjee, Jose Barreiros, Kate Tsui, Tom Silver, and Tapomayukh Bhattacharjee. Prioritouch: Adapting to user contact preferences for whole-arm physical human-robot interaction. *CoRL*, 2025.

[19] Patrick E. McKnight and Julius Najab. *Mann-Whitney U Test*, pages 1–1. 2010. ISBN 9780470479216. doi: https://doi.org/10.1002/9780470479216.corpsy0524. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470479216.corpsy0524.

[20] Marius Memmel, Jacob Berg, Bingqing Chen, Abhishek Gupta, and Jonathan Francis. Strap: Robot sub-trajectory retrieval for augmented policy learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

[21] Austin Narcomey, Nathan Tsoi, Ruta Desai, and Marynel Vázquez. Learning human preferences over robot behavior as soft planning constraints, 2024. URL https://arxiv.org/abs/2403.19795.

[22] Soroush Nasiriany, Tian Gao, Ajay Mandlekar, and Yuke Zhu. Learning and retrieval from prior data for skill-based imitation learning. In *Conference on Robot Learning (CoRL)*, 2022.

[23] Sodtavilan Odonchimed, Tatsuya Matsushima, Simon Holk, Yusuke Iwasawa, and Yutaka Matsuo. Retrieve-augmented generation for speeding up diffusion policy without additional training, 2025. URL https://arxiv.org/abs/2507.21452.

[24] Tom Silver, Rajat Kumar Jenamani, Ziang Liu, Ben Dodson, and Tapomayukh Bhattacharjee. Coloring between the lines: Personalization in the null space of planning constraints. *Under Review*, 2025.

[25] Yen-Ling Tai, Yi-Ru Yang, Kuan-Ting Yu, Yu-Wei Chao, and Yi-Ting Chen. Grits: A spillage-aware guided diffusion policy for robot food scooping tasks, 2025. URL https://arxiv.org/abs/2510.00573.

[26] Siddarth Venkatraman, Shivesh Khaitan, Ravi Tej Akella, John Dolan, Jeff Schneider, and Glen Berseth. Reasoning with latent diffusion in offline reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=tGQirjzddO.

[27] Yanwei Wang, Lirui Wang, Yilun Du, Balakumar Sundaralingam, Xuning Yang, Yu-Wei Chao, Claudia Perez-D'Arpino, Dieter Fox, and Julie Shah. Inference-time policy steering through human interactions, 2025. URL https://arxiv.org/abs/2411.16627.

[28] Han Xue, Jieji Ren, Wendi Chen, Gu Zhang, Yuan Fang, Guoying Gu, Huazhe Xu, and Cewu Lu. Reactive diffusion policy: Slow-fast visual-tactile policy learning for contact-rich manipulation. In *Proceedings of Robotics: Science and Systems (RSS)*, 2025.

[29] Ruolin Ye, Wenqiang Xu, Haoyuan Fu, Rajat Kumar Jenamani, Vy Nguyen, Cewu Lu, Katherine Dimitropoulou, and Tapomayukh Bhattacharjee. Rcareworld: A human-centric simulation world for caregiving robots. *IROS*, 2022.

[30] Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

[31] Guangyao Zhou, Sivaramakrishnan Swaminathan, Rajkumar Vasudeva Raju, J Swaroop Guntupalli, Wolfgang Lehrach, Joseph Ortiz, Antoine Dedieu, Miguel Lazaro-Gredilla, and Kevin Patrick Murphy. Diffusion model predictive control. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=pvtgffHtJm.

[32] A. Zlatintsi, I. Rodomagoulakis, P. Koutras, A. C. Dometios, V. Pitsikalis, C. S. Tzafestas, and P. Maragos. Multimodal signal processing and learning aspects of human-robot interaction for an assistive bathing robot. In *2018 IEEE International Conference on Acoustics,*

Speech and Signal Processing (ICASSP), pages 3171–3175, 2018. doi: 10.1109/ICASSP.2018.8461568.