Learning Visuotactile Grasping Policies from Demonstration Data

Lernen visuotaktiler Greifstrategien aus Demonstrationsdaten
Master thesis in the field of study “Computational Engineering” by Erik Helmut
Date of submission: November 6, 2025

1. Review: Prof. Jan Peters, Ph.D.
2. Review: Niklas Funk, M.Sc.
3. Review: Tim Schneider, M.Sc.
4. Review: Cristiana de Farias, Ph.D.

Darmstadt

Erklärung zur Abschlussarbeit gemäß § 22 Abs. 7 APB TU Darmstadt

Hiermit erkläre ich, Erik Helmut, dass ich die vorliegende Arbeit gemäß § 22 Abs. 7 APB der TU Darmstadt selbstständig, ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt habe. Ich habe mit Ausnahme der zitierten Literatur und anderer in der Arbeit genannter Quellen keine fremden Hilfsmittel benutzt. Die von mir bei der Anfertigung dieser wissenschaftlichen Arbeit wörtlich oder inhaltlich benutzte Literatur und alle anderen Quellen habe ich im Text deutlich gekennzeichnet und gesondert aufgeführt. Dies gilt auch für Quellen oder Hilfsmittel aus dem Internet.

Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mir ist bekannt, dass im Falle eines Plagiats (§ 38 Abs. 2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

Darmstadt, 6. November 2025

E. Helmut

Erik Helmut

Abstract

Robots still struggle to regulate grasp forces in contact-rich manipulation tasks, especially when handling fragile or deformable objects. Existing imitation learning approaches often treat tactile feedback only as an additional observation, leaving applied forces as an uncontrolled consequence of gripper commands. In this work, we introduce a *Force-Aware* imitation learning framework that integrates tactile sensing directly into the action space. Using a GelSight Mini sensor together with the FEATS model for force estimation, we collect demonstration data with a modified UMI gripper. We then deploy the learned policies on a newly developed actuated-UMI gripper with matching geometry. The proposed *Force-Aware* diffusion policy jointly predicts robot pose, grip width, and grip force. At execution time, a dual-mode controller switches between position control of the grip width and closed-loop force control. We evaluate the approach on two tasks with contrasting force demands. The *Force-Aware* diffusion policy achieves higher success rates than *Vision-Only* or *Tactile-Aware* baselines. It also produces force profiles closer to human demonstrations, measured with the Wasserstein distance. These results show that explicitly modeling and controlling forces improves robustness in both high- and low-force scenarios. The framework highlights the role of tactile sensing not just as context but as a control target, enabling more adaptive and contact-aware robotic manipulation.

Zusammenfassung

Roboter haben nach wie vor Schwierigkeiten, Greifkräfte in kontaktintensiven Manipulationsaufgaben zuverlässig zu regulieren, insbesondere beim Umgang mit zerbrechlichen oder verformbaren Objekten. Bisherige Imitationslernansätze nutzen taktiles Feedback meist nur als zusätzliche Beobachtung, wodurch die aufgebrachten Kräfte eine unkontrollierte Folge von Greiferkommandos bleiben. In dieser Arbeit wird ein *Force-Aware* Imitationslern-Framework vorgestellt, das taktiles Feedback direkt in den Aktionsraum integriert. Mithilfe eines GelSight-Mini-Sensors in Kombination mit dem FEATS-Modell zur Kraftschätzung werden Demonstrationsdaten mit einem modifizierten UMI-Greifer erfasst. Die gelernten Steuerungsstrategien werden anschließend auf einem neu entwickelten, aktuierten UMI-Greifer mit gleicher Geometrie ausgeführt. Die vorgeschlagene *Force-Aware* Diffusion Policy sagt Roboterpose, Greifweite und Greifkraft gleichzeitig vorher. Während der Ausführung wechselt ein Dual-Mode-Regler zwischen der Positionsregelung des Greifabstandes und dem geschlossenen Kraftregelkreis. Dieser Ansatz wird an zwei Aufgaben mit unterschiedlichen Kraftanforderungen evaluiert. Die *Force-Aware* Diffusion Policy erreicht höhere Erfolgsraten als die *Vision-Only*- und *Tactile-Aware*-Baselines und erzeugt Kraftprofile, die den menschlichen Demonstrationen näherkommen, gemessen mit der Wasserstein-Distanz. Die Ergebnisse zeigen, dass das explizite Modellieren und Regeln von Kräften die Robustheit sowohl in Szenarien mit hohen als auch mit niedrigen Kräften verbessert. Das Framework verdeutlicht die Rolle des taktilen Feedbacks nicht nur als Kontext, sondern als direktes Regelziel und ermöglicht dadurch eine adaptivere und kontaktbewusstere robotische Manipulation.

Acknowledgments

I would like to express my deepest gratitude to my supervisors, Niklas Funk, Tim Schneider, and Cristiana de Farias, for their guidance, support, and valuable feedback throughout the course of this thesis.

I am also thankful to Prof. Dr. Jan Peters and the Intelligent Autonomous Systems Group for granting me access to the lab facilities and resources, and for fostering an environment that made this work possible.

I also gratefully acknowledge the support of my colleagues and fellow students in the department, whose discussions, collaboration, and encouragement enriched both my research and my overall experience during this work.

In addition, I would like to sincerely thank Niklas Funk and Boris Belousov, whose mentorship during my work as a student assistant and across various projects has profoundly shaped my academic journey and inspired me to pursue further research beyond my master's degree.

Finally, I extend my heartfelt appreciation to my family and friends for their encouragement and unwavering support.

Contents

List of Figures

List of Tables

Abbreviations

BC	Behavioral Cloning
CI	Confidence Interval
CNN	Convolutional Neural Network
DDIM	Denoising Diffusion Implicit Models
DDPM	Denoising Diffusion Probabilistic Models
ECDF	Empirical Cumulative Distribution Function
EE	End-Effector
FEA	Finite Element Analysis
FEATS	Finite Element Analysis for Tactile Sensing
FiLM	Feature-wise Linear Modulation
IRL	Inverse Reinforcement Learning
MDP	Markov Decision Process
PWM	Pulse Width Modulation
ROS	Robot Operating System
UMI	Universal Manipulation Interface

1 Introduction

Humans naturally regulate grasp forces through touch, applying just enough pressure to prevent an object from slipping [27], [26]. This capability relies on rich tactile feedback and adaptive grip force control, and while it comes effortlessly to people, replicating it in robots remains a challenging problem [28], [9]. The selection of an appropriate grasping force has long been recognized as a crucial issue in robotics [5]. Especially when handling fragile or deformable objects, such as fruits or eggs, it is essential to employ the appropriate grasping force to minimize the risk of slippage or breakage. Tactile sensing has emerged as a key technology for safer and more intelligent grasping, as it enables slip detection and inference of shear and normal forces to guide grip force control [14].

Imitation learning has gained traction as a way to transfer human manipulation skills to robots by leveraging demonstrations [38]. Building on this, recent advances have begun to incorporate tactile feedback into robotic grippers. However, in most of these approaches, tactile sensing is treated primarily as an additional observation modality, useful to get additional information for resolving visual occlusion or detecting contact state, but not as a signal that directly shapes the action space [49]. As a result, tactile feedback influences the robot only indirectly through its effect on the observation embedding, while the applied forces themselves remain an uncontrolled consequence of gripper commands.

What remains largely missing is an imitation learning framework in which tactile feedback is not only perceived but also explicitly represented in the action space. Such a formulation would allow the policy to target and regulate the tactile interaction it intends to produce, rather than leaving contact forces as an emergent side-effect of kinematic and gripper control.

In this work, we address this gap by introducing a *Force-Aware* imitation learning framework that integrates tactile feedback directly into the action space. We leverage the GelSight Mini [18], a high-resolution vision-based tactile sensor, together with Finite Element Analysis for Tactile Sensing (FEATS) [21] for estimating normal contact forces during human demonstrations. These demonstrations are collected using an adapted hand-held UMI gripper [11] and then transferred to the actuated-UMI gripper with matching geometry and kinematics for deployment. At execution time, actions are regulated through a dual-mode controller that switches between position control of the grip width and closed-loop force control, depending on whether contact is present or not, ensuring stable behavior across both conditions.

The main contributions of this work are as follows:

- We propose the *Force-Aware* diffusion policy, which predicts robot pose, target grip width, and target grip force jointly, with force represented both in the observation space and as an explicit action, yielding temporally consistent, force-aware action sequences.
- We design and build the actuated-UMI gripper, enabling direct transfer of demonstrations collected with the adapted hand-held UMI gripper equipped with a GelSight Mini sensor.
- We introduce a dual-mode control scheme that mitigates the well-known instability of force control in the presence of discontinuous contacts [8], by switching between position control of the grip width before contact and closed-loop force control once contact is established.
- We evaluate the proposed framework on two contrasting real-world tasks: a high-force *Plant Insertion Task* and a delicate, low-force *Grape Extraction Task*. The results show that explicitly modeling and controlling force improves task success and produces force profiles that are more closely aligned with expert demonstrations than *Vision-Only* or *Tactile-Aware* baselines. In addition to reporting success rates, we also introduce a distributional evaluation metric based on the Wasserstein distance between human and robot force trajectories, providing a more fine-grained measure of similarity.

2 Foundations

Intelligent robotic manipulation comprises the ability to perceive, learn, and generalize across complex physical interactions. This chapter provides an overview of the core concepts underlying this work. Tactile sensing is introduced as a method that enables robots to gather information about physical interactions through touch. Imitation learning is presented as a framework that enables robots to acquire new skills by learning directly from expert demonstrations. Diffusion models are subsequently covered as a class of generative models that have recently been applied to policy learning. Together, these sections establish the background required for the following chapters.

2.1 Tactile Sensing

Our hands are not just tools, they are the critical interface to the world, the connection through which we engage, create, and shape our daily lives. We cannot be responsive and interactive with our environment without being able to feel it [13]. Touch provides certainty and confidence, it speeds up our actions and offers instant feedback on how we are interacting with an object. Unlike humans, robots typically lack a native sense of touch. Replicating this capability in robots, enabling them to truly “feel” helps in understanding the interaction behavior of a real-world object. It provides insights on the object’s weight and stiffness, the surface texture, the deformation upon contact, and its movement under external forces [14].

To date, a wide range of tactile sensors have been developed, including resistive [54], capacitive [47], magnetic [4], as well as vision-based tactile sensors that use either conventional RGB cameras [29], [30], [31], [51] or event-based cameras [15]. In this work, the GelSight Mini tactile sensor [18] is employed to explore how force-aware grasping policies can be learned from human demonstrations.

2.1.1 GelSight Mini Tactile Sensor

The GelSight Mini tactile sensor is a compact, vision-based tactile sensor capable of capturing high-resolution surface details, including the 3D shape and texture of the surface. Its spatial resolution is reported to be well beyond that of human touch [18].

The GelSight Mini operates by using an RGB camera to record the deformation of a soft silicone gel during contact. The gel is mounted on a glass surface and illuminated from three sides using red, green, and blue LEDs (cf. Fig. 2.1). This setup enables photometric stereo, a computer vision technique for estimating the surface normals by observing an object under different lighting conditions [51].

Different gel types are available for the GelSight Mini. Some variants embed visual markers into the gel (cf. Fig. 2.1), thereby improving the ability to track surface deformation and indentation motion by offering more visual features. Others use a clear gel surface without embedded markers.

Compared to other tactile sensing technologies, such as capacitive tactile sensors or force/torque sensors, the GelSight Mini stands out for its simplicity, affordability, and high spatial tactile resolution. The sensor captures images at 25 Hz, enabling real-time usage in closed-loop control tasks. These attributes make the GelSight Mini an appealing entry point into high-fidelity tactile sensing.

However, despite images being rich in visual information, they lack direct, interpretable physical measurements. The raw visual data does not inherently convey metrics such as force or pressure. One proposed solution involves a machine learning approach for the estimation of shear and normal forces from the observed gel deformations.

2.1.2 Force Estimation from GelSight Mini Images

To bridge the gap between raw visual output of the GelSight Mini sensor and physically interpretable quantities, a learning-based framework called FEATS has been proposed [21]. FEATS estimates spatially resolved force distributions directly from raw images captured by the GelSight Mini sensor.

Traditional approaches for extracting force-related information from vision-based tactile sensors often rely on marker displacement methods or reconstructing the surface depth maps via photometric stereo. Nonetheless, these methods are typically limited by the

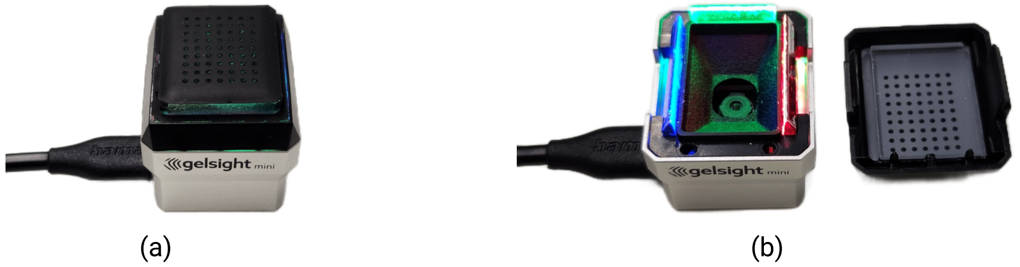


Figure 2.1: The GelSight Mini tactile sensor and its internal components. In (a), the GelSight Mini is shown in its assembled state, as used during operation. With the gel removed in (b), the internal components are revealed. The camera is positioned in the center, surrounded by three LEDs (red, green, and blue) that provide illumination for capturing the surface deformation of the gel.

number of markers or fail to fully account for the nonlinear material behavior of the soft elastomer.

In contrast, FEATS directly learns the mapping from RGB images to shear and normal force distributions using a U-net model [40]. Training data is collected through a series of indentation experiments using a CNC milling machine. This setup enables precise positioning of the GelSight Mini sensor against a variety of 3D-printed indenters. For each real-world indentation, a corresponding Finite Element Analysis (FEA) simulation is conducted using the CalculiX solver (cf. Fig. 2.2).

To accurately model the soft elastomer gel of the GelSight Mini, a hyperelastic Neo-Hookean material model is used in the FEA simulations. The model parameters are calibrated by minimizing the error between simulated forces and measurements obtained from an external force/torque sensor. The calculated underlying force distributions are then projected into the coordinate system of the GelSight Mini, i.e., into an image plane. Within the image boundaries, the force distributions are binned into a fixed-resolution grid to serve as force distribution labels.

The resulting model is trained in a supervised way, taking the raw tactile images as input and predicting the corresponding shear and normal force distributions. In essence, FEATS approximates the output of computationally expensive FEA simulations with a neural network. Unlike FEA, FEATS does not require a precise geometrical description of the contact configuration. This makes FEATS suitable for real-time applications, as only the raw sensor image is required at runtime.

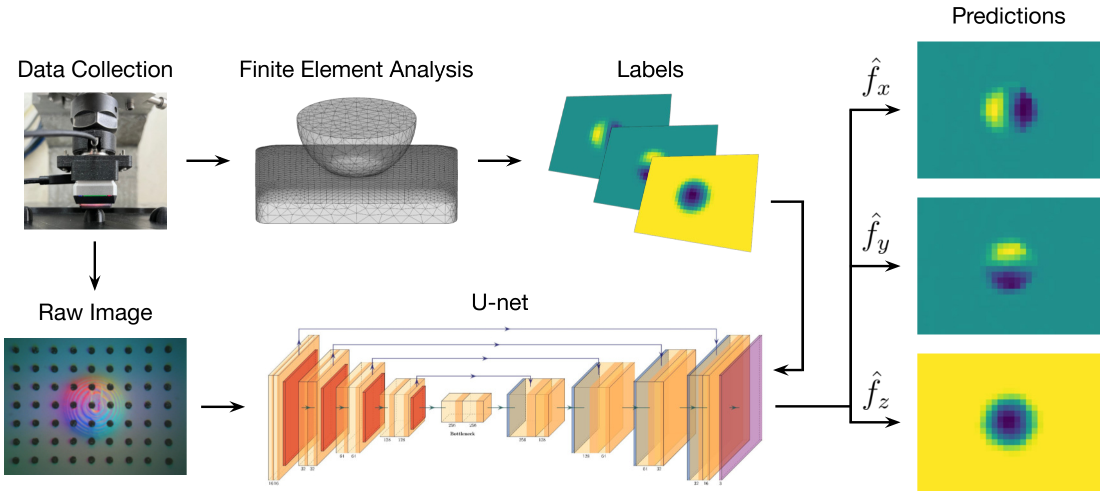


Figure 2.2: FEATS overview from data collection to force distribution prediction. Starting from data collection with a CNC milling machine, FEA is employed for label generation, i.e., calculating the corresponding “ground truth” force distributions. A U-net model is trained to predict shear and normal force distributions from raw tactile images during inference. (Figure from Helmut et al. [21])

FEATS has been shown to generalize well across different indenter shapes and even across different GelSight Mini sensors, making it a robust and practical solution for force estimation in contact-rich robotic tasks.

2.2 Imitation Learning

Imitation is based on one of the most natural and intuitive ways for humans to acquire new skills by observing and copying others. Consider, for example, the process of a young child learning how to fold a paper airplane. The child watches their older sibling execute a series of folding steps, without formal instructions or understanding of why each step works. There are no explicit rules being taught and no external reward is given for success. Over time, through repeated imitation, the child gets better at folding paper planes that actually fly well. The key point is that the child is not learning by being told what to do, nor by evaluating success or failure in a structured way. Instead, the child learns purely by mimicking the observed behavior.

The concept of imitation learning refers to the idea of learning how to solve a task by observing expert demonstrations. Rather than learning through exploration or trial and error, it focuses on reproducing behavior. The objective of imitation learning is to efficiently acquire a desired behavior by imitating an expert's actions [38]. It is a powerful approach for designing autonomous behavior, particularly in robotics and physical systems where manually programming every action or crafting a reward function is difficult or impractical. The reward function is indirectly described by the expert demonstrations.

As described in Osa et al. [38], imitation learning is often formulated in the context of sequential decision making, where the environment is modeled as an Markov Decision Process (MDP). An MDP is a process that satisfies the Markov property, which assumes that the next state s_{t+1} depends only on the current state s_t in a Markov chain. It is defined as a tuple (S, A, P, γ, D, R), where S is a finite set of states, A is the set of actions or control inputs, P is the transition probability function, and $\gamma \in [0, 1)$ is the discount factor that balances short- and long-term rewards. D denotes the initial-state distribution from which the starting state s_0 is drawn, and $R : S \to \mathbb{R}$ defines the reward function that assigns a scalar value to each state.

In the context of imitation learning, however, the reward function R is typically unknown or unused. The goal of imitation learning is to learn a policy π that reproduces the behavior of one or multiple experts who demonstrate how to perform the task. A policy π defines a mapping from states to actions. It can be deterministic, where actions are selected as $a = \pi(s)$, or stochastic, where actions are drawn from a distribution $a \sim \pi(a \mid s)$. In robotics, the learned policy takes the role of a controller by computing the control inputs based on the current state of the system.

Demonstrations from the expert are typically collected as a dataset $\mathcal{D} = \{(s_i, a_i)\}_{i=1}^{N}$, consisting of sequences of state-action pairs. Using the collected dataset $\mathcal{D}$, a common objective of the imitation learning problem is to learn a policy π^* that minimizes the divergence between the expert's behavior and the learned behavior

$$\pi^* = \arg\min_{\pi} \mathrm{Div}(q(\phi), p(\phi)), \tag{2.1}$$

where $q(\phi)$ and $p(\phi)$ represent the distributions over the features induced by the expert's policy and the learner's policy, respectively, and Div denotes a divergence measure between these distributions. Different imitation learning approaches vary in how they define and optimize the policy. The most prominent methods include Behavioral Cloning (BC) and Inverse Reinforcement Learning (IRL). BC [3] treats imitation learning as a supervised learning problem by learning a reactive policy that maps observed states to actions based on expert demonstrations. It does not rely on explicit goals or reward functions, but instead mimics the expert's behavior directly through state-action rules. IRL [37], in contrast, seeks to recover the expert's reward function, which often provides a more compressed description of the behavior. The reward function can then be used within reinforcement learning to learn a policy.

While BC can have difficulty representing multimodal action distributions, IRL is more complex, computationally demanding, and underdetermined because multiple reward functions can explain the same behavior. These limitations have motivated exploration of alternative approaches for imitation learning, including the use of generative models. Building on this idea, recent work on diffusion policies employ diffusion models for visuomotor policy learning [10].

2.3 Diffusion Models for Policy Learning

Diffusion policies frame robot control as a conditional denoising process, gradually transforming noise into action trajectories. These policies have been shown to express multimodal action distributions, maintain robustness when scaling to high-dimensional action spaces, and benefit from stable training. Rather than predicting only the next action, diffusion policies generate sequences of future actions, improving temporal action consistency and avoiding short-sighted planning. These properties have made diffusion policies a popular choice for imitation learning tasks [10].

2.3.1 Denoising Diffusion Probabilistic Models

Denoising Diffusion Probabilistic Models (DDPM), or simply diffusion models, are a class of deep generative models introduced by Ho et al. [22]. The core idea is twofold. First, a forward diffusion process defines a Markov chain that gradually adds small amounts of Gaussian noise to the data in the opposite direction of sampling until the sample is “destroyed”, i.e., becomes indistinguishable from simple isotropic Gaussian noise. Second, a neural network is learned to reverse the diffusion process, which removes noise step by step and reconstructs a clean data sample.

Formally, and following the notation summarized by Chi et al. [10], given a sample $\mathbf{x}^K$ drawn from Gaussian noise, the DDPM performs K denoising iterations to recover the desired noise-free sample $\mathbf{x}^0$. Each reverse diffusion step takes the form

$$\mathbf{x}^{k-1} = \alpha(\mathbf{x}^k - \gamma\varepsilon_\theta(\mathbf{x}^k, k) + \mathcal{N}(0, \sigma^2 I)) \tag{2.2}$$

where ε_θ is the noise-prediction network parametrized by θ, $\mathcal{N}(0, \sigma^2 I)$ is the added Gaussian noise at every step, and α, γ, σ are functions of the iteration step k, referred to as the noise schedule.

Training of the DDPM proceeds by drawing a clean sample $\mathbf{x}^0$ from the dataset, selecting a random denoising iteration step k, and adding noise ε^k to produce a corrupted sample. The DDPM then predicts the added noise. The final objective is a simple mean-squared error:

$$\mathcal{L} = \mathrm{MSE}(\varepsilon^k, \varepsilon_\theta(\mathbf{x}^0 + \varepsilon^k, k)) \,. \tag{2.3}$$

In practice, once trained, diffusion models generate "new" data by starting from random Gaussian noise and applying the learned denoising steps iteratively. Despite their simplicity, these models provides the foundation for diffusion policies.

2.3.2 Diffusion for Visuomotor Policy Learning

Although DDPMs are typically applied to image generation, Chi et al. [10] extend them to the problem of visuomotor policy learning. This requires two central modifications to the original formulation: (i) changing the representation of the output $\mathbf{x}$ from images to robot action sequences and (ii) conditioning the denoising process on the robot's observations $\mathbf{O}_t$. The remainder of this subsection summarizes their approach.

Closed-Loop Action-Sequence Prediction

In robotics, predicting only a single action is often insufficient, as this can result in unstable behavior. To address this, diffusion policies predict entire sequences of actions, thereby encouraging temporal consistency. Specifically, at time step t, the policy processes the most recent T_O observations (observation horizon) and predicts a sequence of T_p actions (prediction horizon). From this sequence, T_a actions (action execution horizon) are executed before re-planning. Typically, T_a is chosen to be smaller than T_p.

This formulation balances temporal consistency and responsiveness. Actions remain coherent over the prediction horizon, while re-planning enables adaptation to unexpected observations. Moreover, it naturally supports receding horizon control [36], where the next prediction is warm-started with the unexecuted actions from the previous prediction, further improving action smoothness.

Conditioning on Observations

Diffusion policies model the conditional distribution $p(\mathbf{A}_t \mid \mathbf{O}_t)$, where $\mathbf{A}_t$ denotes the actions to be predicted and $\mathbf{O}_t$ the current observation history. This contrasts with Janner et al. [24], who model the joint distribution $p(\mathbf{A}_t, \mathbf{O}_t)$. Conditioning the actions directly on observations avoids the need to infer future states, leading to faster inference and more accurate action prediction. Formally, the reverse diffusion step from Eq. 2.2 is modified to

$$\mathbf{A}_t^{k-1} = \alpha(\mathbf{A}_t^k - \gamma\varepsilon_\theta(\mathbf{O}_t, \mathbf{A}_t^k, k) + \mathcal{N}(0, \sigma^2 I)) \tag{2.4}$$

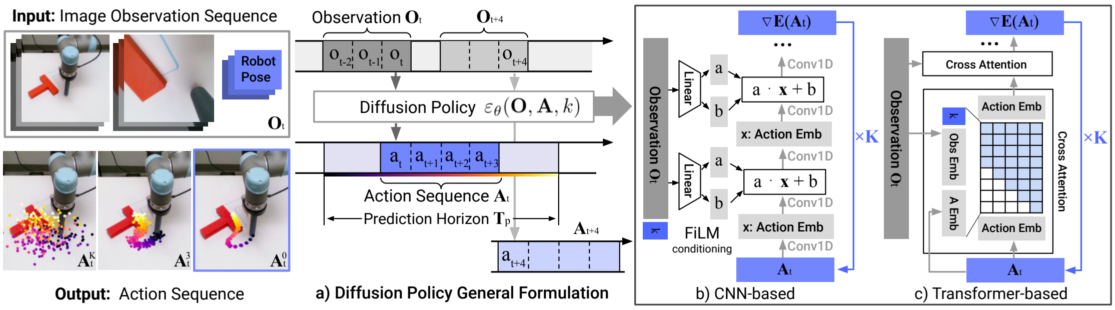
where ε_θ is the noise-prediction network conditioned on both the observations $\mathbf{O}_t$ and current denoising step k. The training loss remains a mean-squared error:

$$\mathcal{L} = \text{MSE}(\varepsilon^k, \varepsilon_\theta(\mathbf{O}_t, \mathbf{A}_t^0 + \varepsilon^k, k)) \ . \tag{2.5}$$

By excluding observations $\mathbf{O}_t$ from the output of the denoising process and using them only as conditioning input, inference becomes significantly faster and more suitable for real-time control. This design also facilitates end-to-end training of the vision encoder.

Network Architecture

Chi et al. [10] evaluate two network architecture types for the noise prediction network ε_θ: Convolutional Neural Networks (CNNs) [41] and Transformers [46] (cf. Fig. 2.3). In this work, we focus exclusively on the CNN-based variant.



Input: Image Observation Sequence
Robot Pose
$\mathbf{O}_t$
$\mathbf{A}_t^K$
$\mathbf{A}_t^3$
$\mathbf{A}_t^0$
Output: Action Sequence
Observation $\mathbf{O}_t$
$\mathbf{O}_{t+4}$
o_{t-2} o_{t-1} o_t
o_{t+4}
Diffusion Policy $\varepsilon_\theta(\mathbf{O}, \mathbf{A}, k)$
a_t a_{t+1} a_{t+2} a_{t+3}
Action Sequence $\mathbf{A}_t$
Prediction Horizon $\mathbf{T}_p$
$\mathbf{A}_{t+4}$
a_{t+4}
a) Diffusion Policy General Formulation
$\nabla\mathbf{E}(\mathbf{A}_t)$
Observation $\mathbf{O}_t$
Linear
a
b
$a \cdot \mathbf{x} + b$
Conv1D
x: Action Emb
$\times$K
k
FiLM
conditioning
$\mathbf{A}_t$
b) CNN-based
Cross Attention
Action Emb
Obs Emb
A Emb
c) Transformer-based

Figure 2.3: Diffusion policy overview. a) General formulation: the policy is conditioned on a sequence of recent observations and predicts a sequence of future actions. b) CNN-based variant: observation features are applied to the network through FiLM conditioning, and the action sequence is generated by iterative denoising. c) Transformer-based variant: in each decoder block, observation features are integrated using multi-head cross-attention. (Figure from Chi et al. [10])

The CNN-based diffusion policy builds on a 1D temporal CNN introduced by Janner et al. [24], with several modifications. Some of these modifications include conditioning action generation on observation features $\mathbf{O}_t$ using Feature-wise Linear Modulation (FiLM) [39] and the current denoising step k. Additionally, only the action trajectory is predicted instead of the concatenated observation-action trajectory, and the goal state conditioning is removed.

For visual encoding, a ResNet-18 network is employed to process raw camera images. Separate encoders are employed for each camera view, and their feature outputs are concatenated to form the observation embedding $\mathbf{O}_t$. This embedding is then applied channel-wise to every convolution layer of the noise-prediction network through FiLM.

3 Related Work

Tactile sensing is a key modality in advancing contact-rich robotic manipulation [6], [14], [13], [33], complementing vision by providing information about forces [21], [16], [42], texture [7], [34], and slip [9], [52], [15]. Tactile feedback is increasingly being integrated into learning-based manipulation frameworks. In particular imitation learning has emerged as a common paradigm where tactile sensing is used to enhance the observation space to improve performance in contact-rich manipulation tasks [17]. Building on this trend, prior work can broadly be divided into two categories: approaches that use tactile feedback as part of the state representation, and approaches that incorporate tactile sensing directly into the action representation.

3.1 Tactile Sensing for State Representation

Most existing work only uses tactile feedback as an additional observation modality to inform actions rather than as a direct control signal. In these approaches, tactile sensing enriches the state space but does not explicitly shape the robot's action representation. Tactile data is often encoded and combined with visual or proprioceptive inputs to improve perception, but the action space remains purely positional or joint-based.

A large part of research focuses on the fusion of visual and tactile data for imitation learning. Works such as 3D-ViTac [23] and GelFusion [25] integrate tactile and visual signals into a unified latent representation, enabling policies to overcome visual occlusion or leverage contact information during manipulation. Similarly, TactileAloha [19] and FreeTacMan [48] demonstrate that tactile features combined with visual observations improve success in contact-rich manipulation tasks compared to vision-only approaches. However, in these systems the policy ultimately outputs standard joint or Cartesian commands, leaving tactile feedback as passive context.

Several studies exploit tactile sensing for contact-aware grasping or failure recovery. For instance, Sharma et al. [43] demonstrate how tactile feedback can mitigate covariate shift in imitation learning by identifying and recovering from failed grasps. Han et al. [20] show how tactile feedback helps predict grasp success and Mao et al. [35] present how tactile feedback helps achieve fine bimanual pinch grasping. Yet again, all attained through richer state embeddings from tactile sensing.

Other approaches emphasize learning from tactile-rich demonstrations. MimicTouch [50] builds on non-parametric imitation learning, where tactile and audio embeddings together with the robot end-effector pose are matched against a demonstration library to retrieve a nearest-neighbor-based action prediction. Online residual reinforcement learning is then used to adapt the policies learned from human demonstrations for robotic execution. Liu et al. introduced ViTaMIn [32], an embodiment-free manipulation interface that integrates vision and custom fin-ray finger tactile sensors into a hand-held gripper based on the Universal Manipulation Interface (UMI) from Chi et al. [11]. They propose a multimodal representation learning strategy to obtain a tactile representation that captures essential contact properties, such as the object's in-hand pose and gripper's deformation. Ablett et al. [1] leverage a see-through visuotactile fingertip sensor attached to the end-effector of a robot to enhance imitation learning via kinesthetic teaching. They introduce tactile force matching, transforming recorded estimated contact forces into replay trajectories using an impedance controller. The resulting robot demonstrations are then used to train policies for contact-rich tasks, such as door-opening. This work conceptually aligns with DexForce [8], which leverages contact forces, measured on a robotic hand with F/T sensors during kinesthetic demonstrations. By converting these forces into force-informed position targets via an impedance controller, the robot can replay the demonstrations, yielding trajectories suitable for policy learning. All these methods highlight the utility of tactile information in demonstrations, but tactile feedback remains an auxiliary signal rather than a directly controlled output.

3.2 Tactile Sensing for Action Representation

While most prior work treats tactile sensing as a passive observation channel, recent efforts have begun exploring its role in shaping the action space itself. Instead of passively conditioning perception, these methods aim to regulate actions through tactile signals, marking an important step toward contact-aware control.

One prominent example is the work from Xu et al. [49], who developed the TactAR teleoperation system to collect demonstration data with real-time visual tactile/force feedback. Building on this data, the authors propose a Reactive Diffusion Policy, where a latent diffusion policy predicts action chunks in latent space at low frequency, while the fast asymmetric tokenizer refines these latent actions at high frequency using real-time tactile feedback, effectively acting as a learned impedance controller. While their method emphasizes closed-loop force control in task space, our approach focuses on local grasp dynamics, where grip force and gripper width are themselves actions predicted by the policy. By embedding forces directly into the action representation, our method provides finer-grained control at the contact interface. Furthermore, while the reactive diffusion policy separates planning and reactive refinement into two subsystems, our single diffusion policy jointly predicts the robot pose, grasp width, and grasp force trajectories, yielding a simpler and more transparent architecture.

The work most conceptually related to ours is by Adeniji et al. [2]. They introduce the Feel-the-Force framework, which also incorporates tactile sensing directly into the action space. Using human demonstrations collected with a tactile glove, their policy predicts gripper end-effector poses together with grasping forces, which are then executed through a PD force controller. This approach, however, relies on a calibrated setup and a reset alignment between the human hand and robot gripper, as well as manual annotation of semantic keypoints to initialize scene representations. Moreover, their execution is constrained by binarized gripper states and requires the force controller to converge before the robot advances to the next action, which slows execution and limits adaptability. By contrast, our method learns directly from robot-embodied demonstrations through a hand-held gripper, avoiding cross-domain retargeting. Instead of a binary gripper state, we predict continuous grip width and target forces, enabling smoother and more precise control. A dual-mode controller scheme decides when to regulate grip width versus grip force, allowing execution to proceed without waiting for force convergence and enabling adaptation to real-time tactile feedback.

Together, these works demonstrate the promise of incorporating tactile sensing into the action space. Yet, they either remain task-space oriented or rely on rigid execution protocols. Our approach advances this direction by directly coupling continuous grip force control with diffusion policy learning from robot-embodied tactile data, enabling more adaptive and contact-aware manipulation.

4 Methods

In this work, we introduce a *Force-Aware* imitation learning framework to learn grasping policies from demonstration data. To capture detailed contact interactions during manipulation, we integrate the GelSight Mini sensor into a custom-built robotic gripper, enabling high-resolution tactile feedback at the fingertip. From the sensor's raw tactile images, we extract estimates of the applied contact forces using Finite Element Analysis for Tactile Sensing (FEATS) (cf. Sec. 2.1.2). This force information is integrated as both state and action in a diffusion policy, which is trained to replicate human demonstrations not only in terms of gripper motion, but also by explicitly predicting and controlling the target grip force applied to the object. This framework is designed to generalize across tasks requiring either strong or delicate manipulation. The following sections describe the gripper hardware, data collection and processing pipeline, the design of the *Force-Aware* diffusion policy, and the implementation of closed-loop force control for deployment on a real robot.

4.1 Gripper Hardware

Our approach relies on two closely related grippers: an adapted Universal Manipulation Interface (UMI) gripper for demonstration data collection, and a custom-built actuated-UMI gripper for robotic deployment (cf. Fig. 4.1).

For demonstrations, we use a modified version of the UMI gripper [11], a hand-held gripper designed to allow direct transfer of human manipulation skills to robotic systems. Our modified version of the UMI gripper replaces the original GoPro camera with an Intel RealSense D405, which provides in-hand RGB images and is tracked via OptiTrack[1] markers for precise motion capture. The standard elastic TPU fingers are replaced with

[1] OptiTrack is a real-time motion capture system with precise 6DoF tracking: `https://optitrack.com`

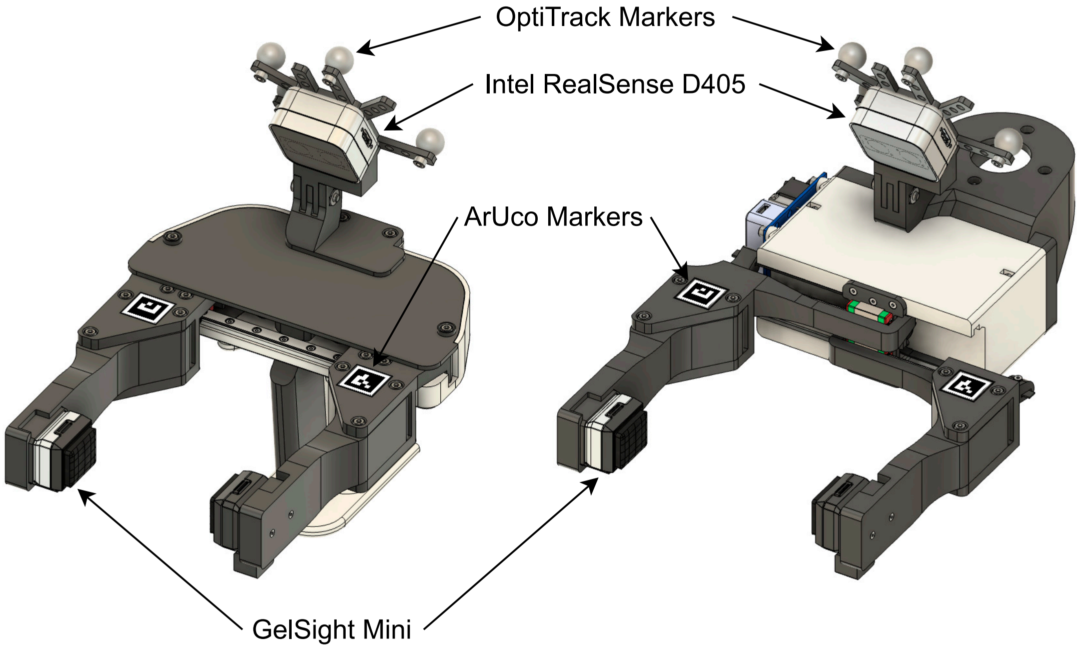


Figure 4.1: Side-by-side comparison of the adapted UMI gripper (left) used for demonstration data collection, and the actuated-UMI gripper (right) used for robotic deployment. Both designs feature an Intel RealSense D405 camera, a GelSight Mini tactile sensor (mounted on one fingertip), OptiTrack markers for motion tracking, and ArUco markers for grip width measurement. Sensor placement and overall geometry are matched to enable direct transfer of learned policies to the actuated-UMI.

rigid fingers. One finger is fitted with a GelSight Mini sensor at its fingertip, and the other finger holds a shell of the GelSight Mini sensor with a matching gel pad but no electronics. In addition, we attach ArUco[2] markers to each finger and use the in-hand Intel RealSense D405 camera to track their positions, enabling precise measurement of the grip width.

To deploy learned policies on a real robot, we developed the actuated-UMI gripper[3].

[2] ArUco markers are square markers consisting of a wide black border and an inner binary matrix that determines their identifier: `https://docs.opencv.org/3.4/d5/dae/tutorial_aruco_detection.html`

[3] The Actuated UMI Gripper is a fully 3D-printable, open-source, modular, and cost-efficient robotic gripper based on the original UMI design: `https://actuated-umi.github.io`

Powered by a single DYNAMIXEL XL430-W250-T motor, this gripper uses a belt-driven mechanism to synchronously actuate both fingers. Two coil springs, one per finger, return the gripper to its open position when torque is released. The geometry mirrors the adapted hand-held UMI gripper. All sensors and markers, including the GelSight Mini, the Intel RealSense D405, the OptiTrack markers, and the ArUco markers are positioned identically, enabling seamless transfer of policies learned on demonstration data. The actuated-UMI supports multiple control modes, such as position, velocity, Pulse Width Modulation (PWM) and operates at approximately $50\,\mathrm{Hz}$, making real-time force control feasible.

4.2 Data Collection

By using the adapted hand-held UMI gripper, we are able not only to demonstrate the required motions, but also to directly apply the necessary contact forces for each task. Capturing both gripper kinematics and high-resolution tactile feedback during demonstrations enables us to record the force profiles essential for learning policies that require precise grip control, something not achievable with conventional teleoperation or kinesthetic teaching methods (cf. Fig. 4.2).

During each demonstration, we record synchronized streams of all relevant sensor data:

- **RGB images** from the Intel RealSense D405 camera $(848 \times 480 \times 3)$.
- **GelSight Mini tactile images** $(320 \times 240 \times 3)$.
- **Gripper pose** via OptiTrack, tracking the rigid body frame defined by the marker constellation.
- **Grip width** from the ArUco markers on the fingers, with positions determined by the in-hand Intel RealSense D405 camera.
- **Force distribution estimates** computed from each GelSight Mini image using the FEATS model.

For the gripper pose, we record the position as 3D coordinates and the rotation as a 6D feature representation [53], avoiding ambiguities in orientation encoding. All sensor data are acquired using the Robot Operating System (ROS), stored in rosbags with reliable communication settings (reliability: `reliable`, history: `keep_all`, depth: `10`) to prevent data loss.

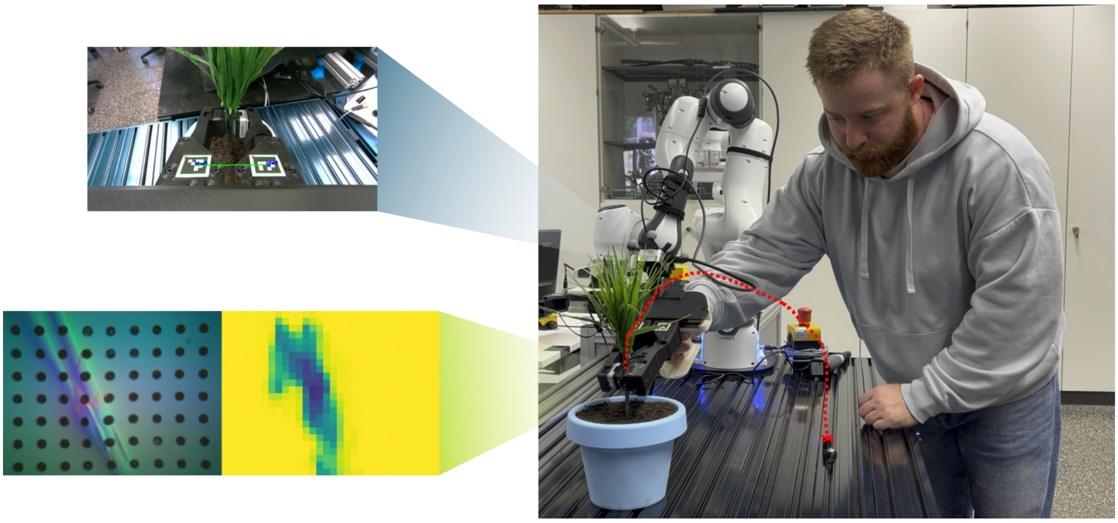


Figure 4.2: Demonstration setup using the adapted hand-held UMI gripper. Right: An expert performs a task using the adapted hand-held UMI gripper. Top left: In-hand RGB camera view with ArUco markers for grip width measurement. Bottom left: GelSight Mini tactile image and corresponding FEATS normal force estimate, visualizing the contact interaction and force distribution during demonstration.

After recording, we synchronize all sensor streams to the GelSight Mini images, which operates at 25 Hz and act as the reference clock. For each GelSight Mini timestamp, we associate the temporally closest RealSense D405 image, OptiTrack and grip width measurement with a tolerance of 0.4 s. This ensures that every sample in the trajectory contains a complete set of sensor observations. The force estimations from FEATS are already timestamped identically to the corresponding GelSight Mini images and thus require no further alignment.

In practice, sensor data loss is rare, but occasional missing grip width values can occur if ArUco markers are temporarily occluded or overexposed. In such cases, missing values are linearly interpolated from nearby valid measurements to maintain a complete time series.

We trim each demonstration to exclude irrelevant data before initial contact and after task completion. For this, we use a threshold of -0.5 N on the estimated total normal force. To account for preparatory and finishing actions, we extend the cropped window by three seconds at both the start and end of each trajectory. This standardizes the sequence boundaries while preserving all relevant manipulation data. Finally, we subsample each demonstration by retaining every fourth data point. This reduces the dominance of idle or static periods in the dataset, pushing the training set to represent more meaningful action phases.

4.3 Force-Aware Diffusion Policy

We build on the diffusion policy introduced by Chi et al. [10], using the 1D temporal CNN variant with FiLM conditioning [39] (cf. Sec. 2.3.2). To incorporate tactile feedback into the diffusion policy in a physically meaningful and interpretable way, we extend the observation and action spaces beyond those used in prior work [10], [11]. Each policy input at time t includes:

1. **An in-hand RGB image** from the Intel RealSense D405 camera, downsampled to $96 \times 96 \times 3$ for computational efficiency without sacrificing the essential visual context.
2. **The gripper pose**, consisting of 3D position coordinates and 6D rotation feature representation [53] providing an unambiguous and continuous encoding of spatial orientation.
3. **Grip width**, calculated as the Euclidean distance between the centers of the ArUco markers on the two fingers, measured in the Intel RealSense D405 image.
4. **Tactile feedback** is represented by a force estimate, which we extract from each GelSight Mini tactile image using a pretrained and fixed FEATS model. Although FEATS outputs a spatial distribution of normal force, we integrate over the discretized force distribution to obtain the total normal force. This scalar value is used, as it directly corresponds to the quantity regulated during closed-loop force control on the gripper.

The diffusion policy is designed to predict action trajectories consisting of the absolute target pose, target grip width g_d, and target grip force f_d, each over a fixed prediction horizon of 32 and action execution horizon of 16. The observation horizon comprises the two most recent observations, which allows the policy to capture short-term context for decision-making (cf. Fig. 4.3). By incorporating both force and grip width as elements of the state and action space, these quantities are used for conditioning the model as well as being treated as output variables during the denoising process. This design ensures

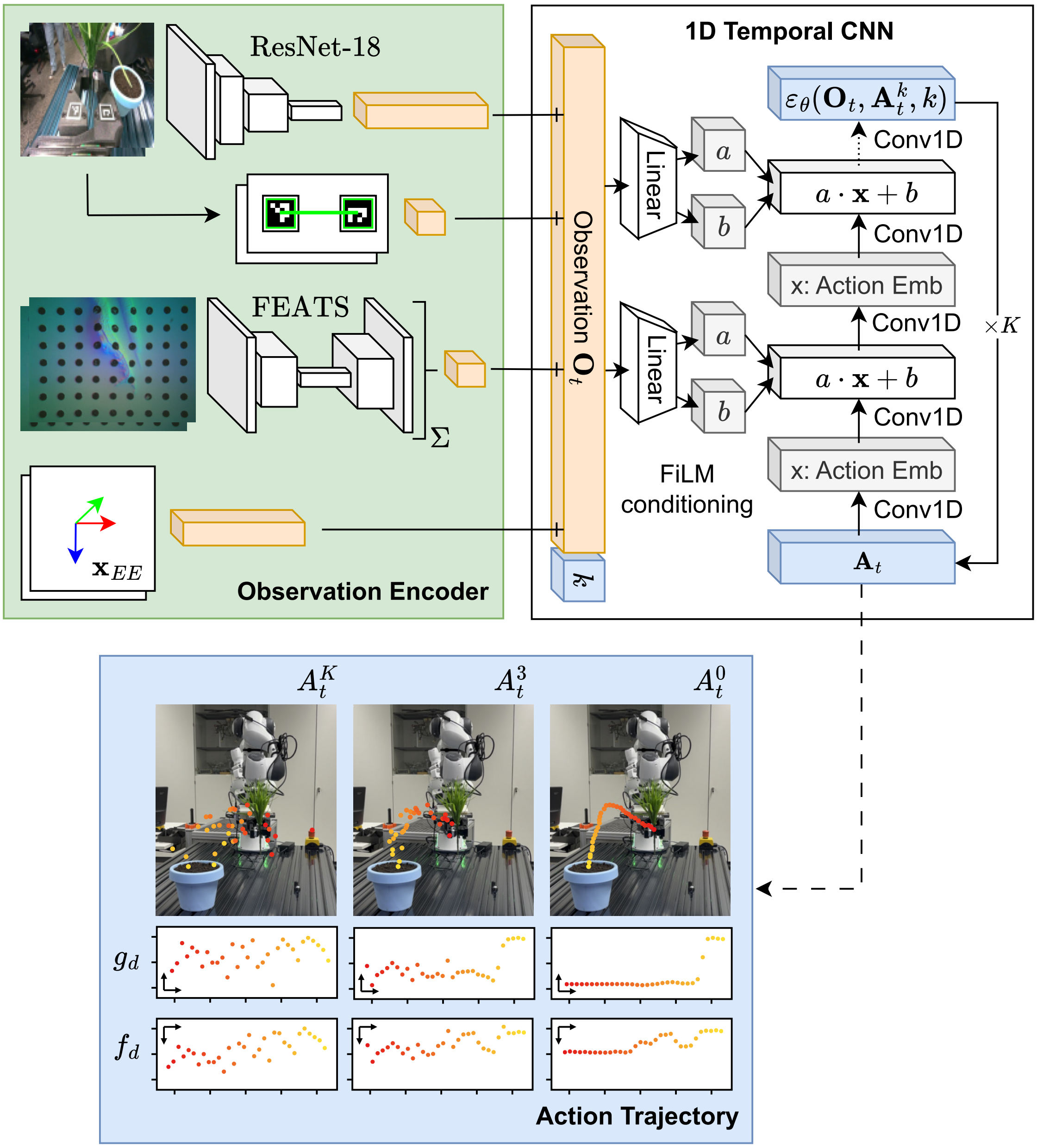


Figure 4.3: Schematic of the *Force-Aware* diffusion policy architecture. Visual, proprioceptive, and tactile observations are encoded and provided as input to a 1D temporal CNN with FiLM conditioning. The model predicts action trajectories including absolute end-effector pose, grip width, and grip force. This structure enables closed-loop force control of the gripper during manipulation.

that the current force and grip width observations directly influence the predicted action trajectory, including the target force and target grip width at each step. As a result, the model's predictions align with current observations, mitigating the risk of implausible or unstable grasping behavior. This enables the policy to anticipate and regulate the amount of force required for subsequent steps.

The policy is trained using a mean squared error loss on the noise added during the denoising process. We use the publicly available implementation of the diffusion policy from LeRobot[4] with modifications to support our extended observation and action modalities.

4.4 Policy Deployment and Gripper Control

Including both target grip width and target force in the action space allows us to capture both positional and force-related aspects of manipulation. Target grip force is the relevant control variable during object contact because it enables closed-loop force control. However, target grip width is required to guide finger positioning during phases when there is no contact, such as when approaching, grasping, or releasing an object. Without explicit grip width actions, the policy would lack the means to open or close the gripper accurately outside the contact phase. This dual-action design reflects how humans intuitively adjust both finger placement and applied force during object manipulation.

To deploy learned policies on the actuated-UMI gripper, we therefore implement a dual-mode control strategy that switches between grip width control and force control based on the current interaction phase (cf. Fig. 4.4). This strategy allows the robot to execute both pre-contact motions and in-contact, closed-loop force control. The diffusion policy outputs both a target grip width g_d and a target force f_d at each step. The controller monitors both the target force and the estimated contact force $\hat{f}$, computed from FEATS using the latest GelSight Mini image I_{GS}. If both the target and estimated force are below $-0.5\,\mathrm{N}$, the system assumes that the robot is in contact with the object and switches to force control. The switching threshold of $-0.5\,\mathrm{N}$ was selected based on the noise characteristics of the FEATS model, ensuring that the controller only transitions to force control when actual contact is confidently detected. The controller computes the force error $e = \hat{f} - f_d$ and applies a PID controller to this error:

[4] LeRobot is a Hugging Face community providing models, datasets, and tools for real-world robotics in PyTorch: `https://github.com/huggingface/lerobot`

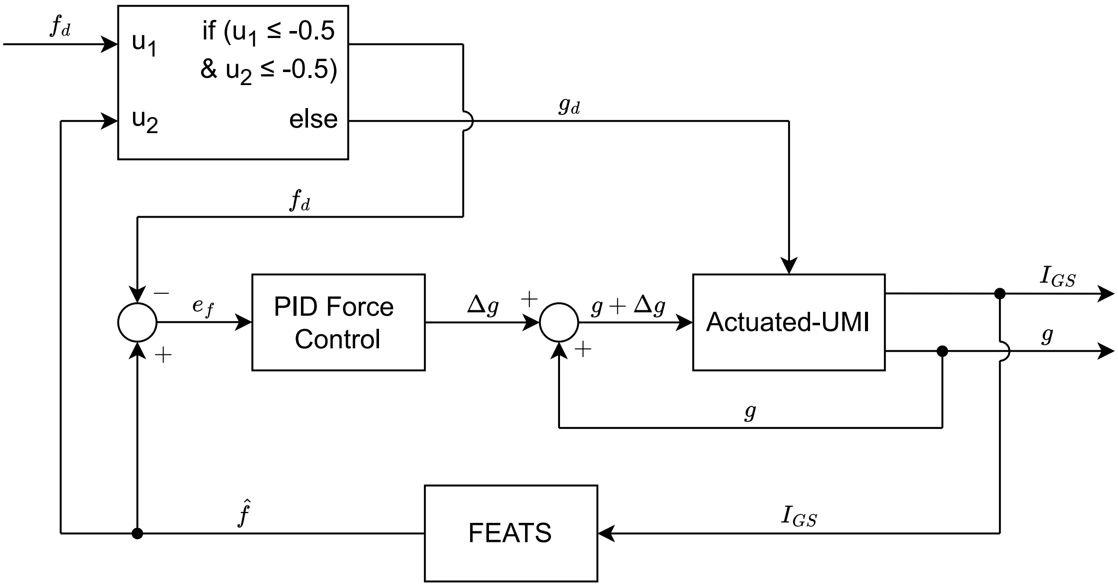


Figure 4.4: Control flow diagram for dual-mode gripper control. When both target and measured force are below -0.5 N, the controller switches to PID force control using FEATS force feedback. Otherwise, the controller uses direct position control based on the target grip width. All actions are mapped to motor position commands for the actuated-UMI gripper.

$$\Delta g = K_P e(t) + K_I \int e(t)dt + K_D \frac{d}{dt} e(t) \tag{4.1}$$

where Δg is the incremental grip width adjustment. This value is added to the current grip width and sent as a position command to the internal PD controller of the DYNAMIXEL motor. The force control loop runs at 25 Hz, synchronized with GelSight Mini image acquisition and FEATS prediction. During deployment, the diffusion policy runs at 7 Hz, while the lower-level force and position controllers operate at higher rates to bridge the gap between high-level action selection and real-time motor actuation. If either the target or measured force is above -0.5 N, the controller assumes the robot is not in contact and directly sends the target grip width g_d to the internal PD controller of the DYNAMIXEL motor.

A PID controller is chosen for force control to balance responsiveness and stability. A small integral gain is included to reduce steady-state errors that can arise in phases where the force does not vary rapidly, helping to eliminate offsets that a pure PD controller may leave uncorrected. Output limits with anti-windup are applied to ensure the integral term cannot accumulate beyond actuator limits. Controller gains and output limits are tuned based on observed system performance.

To further ensure seamless transfer from demonstration data to robot execution, two calibrations are performed. First, hand-eye calibration is carried out using the method of Tsai and Lenz [44], aligning the OptiTrack world frame with the robot base. This transformation allows us to directly command end-effector positions to the robot in its own base frame during policy execution. Second, a linear mapping between grip width, measured as the ArUco marker distance, and motor position is estimated via least squares by slowly closing the actuated-UMI gripper and recording both quantities. This ensures that grip width values predicted by the policy can be accurately converted into motor commands during deployment.

5 Experiments

In this chapter, we present the experimental evaluation of our proposed method using two robotic manipulation tasks with differing physical interaction demands. The experiments focus on comparing input modalities to the diffusion policy, particularly the impact of incorporating tactile feedback. We evaluate both *task success* and *imitation quality*. The following sections describe the experimental design and detail the experimental setup and results for each task.

5.1 Experimental Design

We evaluate our method using two distinct robotic manipulation tasks that cover a range of force requirements. The *Plant Insertion Task* involves high-force contact to insert a plastic plant into real soil, while the *Grape Extraction Task* requires delicate, low-force manipulation to remove a real grape from a toothpick without damaging it. In the following, we describe the choice of input modalities, baseline configurations, general experimental setup, and evaluation metrics used for the experiments.

5.1.1 Input Modalities and Baselines

All trained diffusion policies receive multimodal input, combining proprioceptive[1] and exteroceptive[2] signals to support both coarse and fine-grained manipulation. The input always includes the robot's End-Effector (EE) pose and grip width, as well as the RGB image from the Intel RealSense D405 camera mounted on the actuated-UMI gripper (cf. Sec. 4.1). The key variable among the evaluated models is the use and format of tactile

[1] Proprioception is the perception of the body's position and movement [45].

[2] Exteroception refers to sensing external stimuli such as sight, smell, hearing, touch, and taste [45].

feedback from the GelSight Mini sensor. In this section, we evaluate three strategies that differ in how (or whether) tactile sensing is used. Our goal is to understand how tactile information contributes to task success and force-aware behavior. For simplicity, the following descriptions focus only on the differences in tactile sensing and gripper-related state and action modalities. Other input and output components of the diffusion policy remain consistent across all strategies and are described in Sec. 4.3. Each policy is trained for 60000 steps. A summary of the different strategies can be found in Tab. 5.1.

Force-Aware Strategy	
Input Modalities	Vision, EE pose, grip width, tactile (force)
Action Space	Target EE pose, target grip width, target force
Force Control Strategy	Explicit force control (closed-loop via estimated force)
Tactile-Aware Strategy	
Input Modalities	Vision, EE pose, grip width, tactile (image)
Action Space	Target EE pose, target grip width
Force Control Strategy	No force control (reactive to tactile image)
Vision-Only Strategy	
Input Modalities	Vision, EE pose, gripper state (open/closed)
Action Space	Target EE pose, gripper command (open/close)
Force Control Strategy	No force control (open-loop in force space)

Table 5.1: Comparative summary of the proposed method and baselines with respect to input modalities, action space, and force control strategy. While all policies share visual and proprioceptive inputs, they differ in whether and how tactile sensing is incorporated. These differences influence both the structure of the action space and how the gripper can be controlled.

Proposed Method: Force-Aware Strategy

Our primary method uses force estimations from the pretrained FEATS model as tactile input. The diffusion policy receives the total normal force and absolute grip width as inputs and predicts the target force and target grip width. Incorporating a single force signal results in an interpretable value that is used during both state encoding and action generation. This allows for closed-loop force control, in which the robot explicitly controls the force it applies to an object.

Using a dual-mode controller, this approach distinguishes between a no-contact phase, where only the gripper width is controlled, and a contact phase, where the interaction force is actively regulated. This distinction is critical for fine manipulation and handling delicate objects. Without reliable contact detection and an explicit target force, the policy would lack a reference for closed-loop force control. See Sec. 4.3 and Sec. 4.4 for a more detailed description of our proposed method.

Baseline 1: Tactile-Aware Strategy

In this baseline, we provide the diffusion policy with the raw tactile image from the GelSight Mini alongside the input from the RGB camera. Both are resized to $96 \times 96 \times 3$ and processed using separate visual encoders. The policy neither receives an explicit force signal nor predicts a target force. Rather, it only predicts the target grip width, i.e., the desired finger distance. Absolute grip width is used instead of relative motion to prevent prediction error accumulation over longer time horizons.

With this setup, tactile sensing is used more passively, primarily to estimate contact state, but not to perform explicit force control. The robot implicitly reacts to tactile feedback through visual features. Accordingly, this policy cannot actively control contact forces, though it may still benefit from situational awareness.

Baseline 2: Vision-Only Strategy

As a minimal baseline, we evaluate a diffusion policy that receives no tactile input. This policy operates using proprioception and vision only. Rather than using the continuous absolute grip width as part of the state and action, we convert the gripper command into a binary signal that indicates whether the gripper should be open or closed. This binary signal is derived from the same demonstrations used for the other policies by applying a threshold to the grip width. If the grip width falls below a predefined threshold, the gripper state is labeled as closed $(= 1)$, and otherwise it is considered open $(= 0)$.

During inference, the policy only predicts whether to open or close the gripper. The internal PD controller of the motor inside the actuated-UMI enforces this command by moving the fingers to a position where the gripper is fully closed (i.e., the fingers touch).

Since the internal PD controller lacks an integration term, the fingers will continue to close until the motor cannot overcome the mechanical resistance. This results in unmodulated, coarse force application.

5.1.2 General Experimental Setup

All experiments are conducted using a real Franka Research 3 robot equipped with the actuated-UMI gripper (see Sec. 4.1). The robot is controlled using Cartesian position control from franky[3]. During inference, the robot moves with a relative velocity of 2%. Across all strategies, actuation of the actuated-UMI gripper relies on the internal PD position controller of its DYNAMIXEL motor. The motor is controlled through the dynamixel-api[4]. We use the Robot Operating System (ROS) for sensor data acquisition and synchronization. Sensors and the policy run in separate ROS nodes, with the policy operating at 7 Hz.

For both tasks, we collected around 30 demonstrations using the UMI gripper (see Sec. 4.2), all performed by the same expert. All diffusion policies are trained with an observation horizon of 2, prediction horizon of 32, and action execution horizon of 16. We train the diffusion policies using Denoising Diffusion Probabilistic Models (DDPM) with 100 denoising steps. For inference, we adopt Denoising Diffusion Implicit Models (DDIM) with 10 steps to reduce the number of sampling iterations [10].

Each strategy is evaluated on 20 rollouts per task, across 4 different starting gripper orientations to test generalization under varied initial conditions. For both tasks, object and target positions were fixed throughout all demonstrations and rollouts.

5.1.3 Evaluation Metrics

We evaluate task performance using two complementary criteria: *task success* and *imitation quality*. Task success is measured as the success rate computed across all evaluation rollouts for a given strategy. Imitation quality is assessed by measuring how closely the force patterns obtained via FEATS from a strategy's learned policy trajectories match those from the demonstrations. This similarity is measured using several statistical metrics as well

[3] franky is a high-level control library for the Franka robots: `https://github.com/TimSchneider42/franky`

[4] dynamixel-api is a Python wrapper for the DYNAMIXEL SDK library, designed to control various DYNAMIXEL motors: `https://github.com/TimSchneider42/dynamixel-api`

as the Wasserstein-1 distance, all of which are computed with an equal-mass weighting scheme.

Equal-Mass Weighting of Trajectories

Since trajectories differ in duration, pooling all samples would bias the statistics toward longer trajectories. To ensure each trajectory contributes equally, we assign every trajectory the same total weight and distribute that weight uniformly over its samples. For a set of M trajectories with lengths n_m, each sample in trajectory m receives a weight

$$w_{m,k} = \frac{1}{M \cdot n_m} \, . \tag{5.1}$$

Statistical Summaries

For each strategy's rollouts and for the demonstrations, we compute the following statistics using the equal-mass weighting scheme:

Weighted mean. The average applied force is computed as

$$\mu = \sum_i w_i x_i \, , \tag{5.2}$$

where x_i is a force sample and w_i is its assigned weight.

Weighted standard deviation. The variability under the same weighting scheme is computed as

$$\sigma = \sqrt{\sum_i w_i x_i^2 - \mu^2} \, . \tag{5.3}$$

Weighted median. This is the force value at which the cumulative sum of weights reaches 0.5, meaning that half of the total trajectory-normalized weight lies below and half above this value.

Minimum force. This is reported as the largest negative observed force value, independent of weighting.

Wasserstein-1 Distance

To quantify the difference between the force distributions of the demonstrations and rollouts, we use the Wasserstein-1 distance W_1, also known as the Earth mover's distance [12]. For two 1D probability mass functions u and v, it is defined as

$$W_1(u,v) = \inf_{\pi \in \Gamma(u,v)} \int_{\mathbb{R}\times\mathbb{R}} |x-y| d\pi(x,y) \tag{5.4}$$

where $\Gamma(u,v)$ denotes the set of all joint distributions with marginal u and v. In our case, u and v are the weighted empirical distributions of force values from the demonstrations and from the rollouts of a given strategy, using the equal-mass weighting scheme described earlier.

The W_1 value corresponds to the "average amount" the force distributions from one set would need to be shifted to match the other, expressed in the same unit as the force (N). Smaller values therefore indicate that the forces measured during rollouts are more similar to those of the demonstrations, while larger values indicate greater differences in either magnitude or distributional shape.

5.2 Plant Insertion Task

In the *Plant Insertion Task*, the robot must grasp a plastic plant from a fixed start position and insert it into a flower pot filled with real soil. After successful insertion, the robot releases the plant. The main challenge is applying sufficient grip force to insert the plant without losing grip. This task serves as a high-force manipulation scenario.

5.2.1 Experimental Setup

Fig. 5.1 shows the experimental setup used for the *Plant Insertion Task*. The pot and plant are kept at fixed positions on the workspace. Before each rollout, the soil is moistened to increase cohesion and improve grip, helping the plant remain firmly in place after insertion.

To standardize the initial conditions, the robot's movement at the beginning of each rollout is constrained. Specifically, the policy initially controls only the gripper. Once a grasp is detected, the robot performs a predefined upward motion to take the plant from its holder before giving full control to the policy.

Grasp detection is strategy-dependet:

- In the *Force-Aware* strategy, a grasp is assumed if the estimated normal force exceeds a small threshold.
- In the *Tactile-Aware* strategy, grasp detection is triggered when the image difference between a reference GelSight Mini image (pre-contact) and the current frame exceeds a threshold.
- In the *Vision-Only* strategy, grasp is assumed when the grip width falls below a closing threshold.

A rollout is considered successful if the plant remains upright in the soil without tilting, falling, or touching the pot rim.

5.2.2 Experimental Results

Across 20 rollouts, the *Force-Aware* strategy achieves the highest success rate of 95% (cf. Fig. 5.2). It consistently inserts the plant while maintaining a firm grip and ensuring that the plant is standing properly in the soil. A successful rollout using the *Force-Aware* strategy can be seen in Fig. 5.4. The *Vision-Only* baseline achieves a moderate success rate of 85%, while the *Tactile-Aware* baseline performs the worst with a success rate of 65%. Failures mainly occur when insufficient force is applied during travel or insertion, leading to slippage or incomplete planting. This is particularly common in the *Tactile-Aware* baseline since there is no direct force control.

Tab. 5.2 summarizes the weighted force statistics and Wasserstein-1 distances between demonstrations and the three strategies. The demonstrations exhibit consistently high

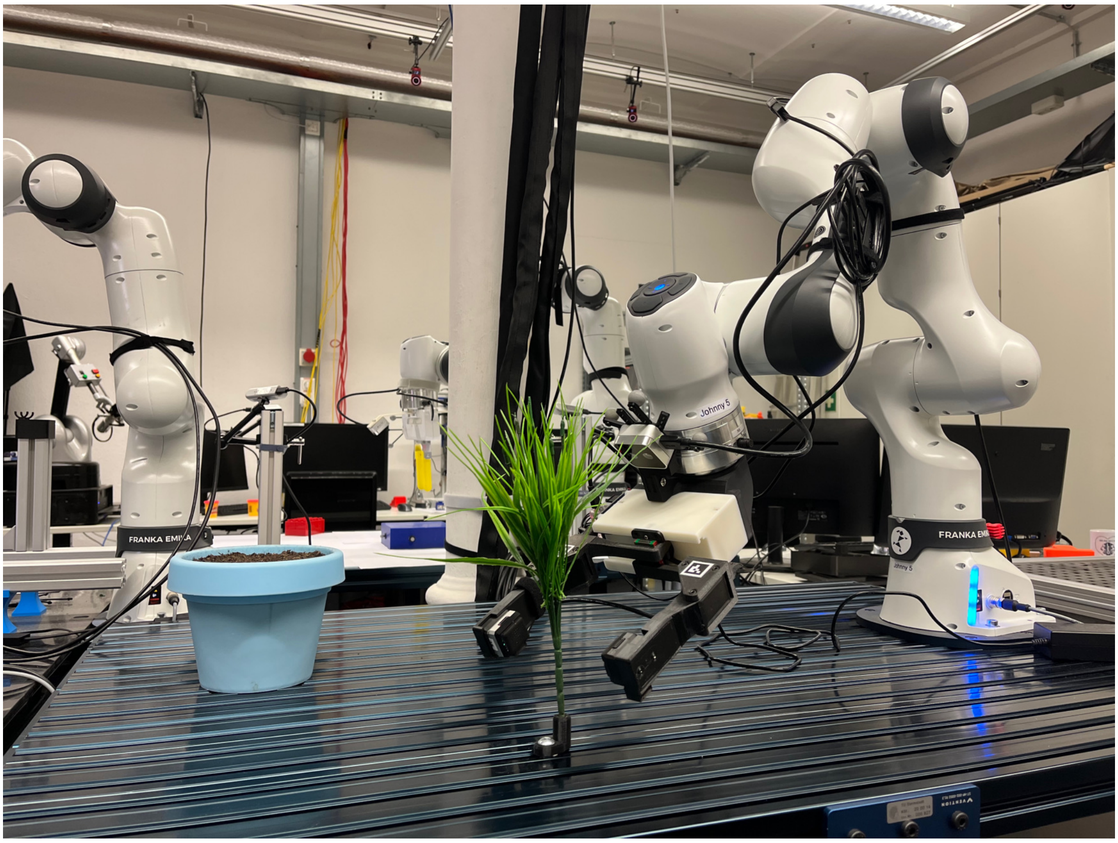


Figure 5.1: Experimental setup for the *Plant Insertion Task*. The image shows the Franka Research 3 robot equipped with the actuated-UMI gripper in its initial state before starting a rollout. The robot must grasp the plastic plant from its holder and insert it into the blue pot filled with real soil.

grip forces with a mean of $-8.4\,\mathrm{N}$ and a narrow spread, as described by the standard deviation of $2.4\,\mathrm{N}$. The *Force-Aware* strategy reproduces the overall magnitude of the demonstrations most closely, with a mean of $-6.5\,\mathrm{N}$ and a Wasserstein-1 distance of $1.92\,\mathrm{N}$. The 95% Confidence Interval (CI) of the Wasserstein-1 distance, obtained via 1000 bootstrap resamples, ranges from $[1.04\,\mathrm{N}, 2.95\,\mathrm{N}]$, indicating robust similarity between demonstrations and *Force-Aware* rollouts despite some variability. The *Vision-Only* baseline also approaches the demonstration force range but with slightly larger deviation shown by the Wasserstein-1 distance of $1.99\,\mathrm{N}$. The *Tactile-Aware* strategy deviates strongly, applying significantly lower forces on average $-3.4\,\mathrm{N}$ and showing the largest Wasserstein-1 distance of $5.07\,\mathrm{N}$, reflecting a clear mismatch in the force distribution.

To complement these summary metrics, Fig. 5.3 visualizes the weighted Empirical Cumulative Distribution Functions (ECDFs) and histograms for demonstrations and rollouts. The Wasserstein distance is directly related to the area between the two ECDF curves. Closer curves indicate higher similarity, while shifts or gaps reveal systematic differences. In our results, the *Force-Aware* and *Vision-Only* strategies produce ECDFs that lie closer to the demonstrations, while the *Tactile-Aware* curve is clearly shifted toward weaker forces. The overlapping histograms confirm this observation, showing that *Tactile-Aware* rollouts rarely reach the higher grip forces seen in the demonstrations.

These differences reflect the underlying control principles. With closed-loop force control, the *Force-Aware* strategy can actively target and maintain a firm grip force, enabling stable insertion of the plant into the soil. In contrast, the *Tactile-Aware* strategy can only use tactile feedback passively to detect contact without controlling the applied force. As a result, it often falls short of the grip force demonstrated by the human, leading to frequent failures. The *Vision-Only* baseline performs reasonably well in this high-force setting because its binary open/close command effectively clamps the fingers together with substantial force. Although the gripper is coarsely controlled, the force values happen to be close to those seen in the demonstrations.

This correspondence shows that success in a high-force task depends directly on matching the demonstrated force profile, with insufficient grip force leading to failure.

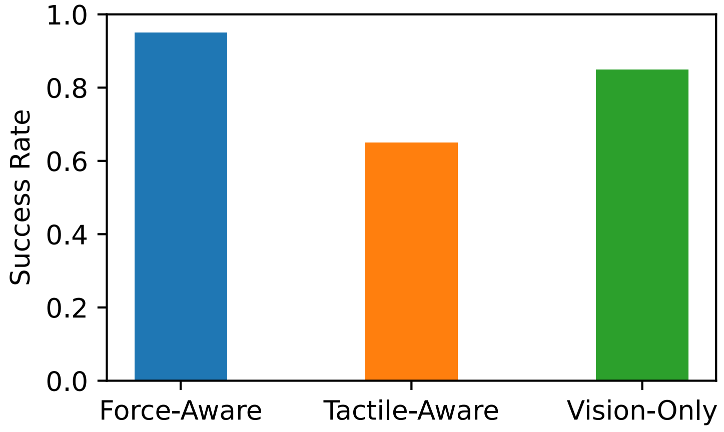


Figure 5.2: Success rates for the *Plant Insertion Task*. The *Force-Aware* strategy achieves the highest success rate at 95%, followed by the *Vision-Only* strategy at 85%, and the *Tactile-Aware* strategy at 65%.

Metric (N)	**Demo**	**Force-Aware**	**Tactile-Aware**	**Vision-Only**
μ	-8.4247	-6.5057	-3.3523	-6.4455
σ	2.3892	2.8899	3.0545	3.2459
$\tilde{x}$	-8.5755	-5.8804	-3.4873	-6.6978
min	-16.2255	-17.4085	-10.3348	-17.0829
W_1	–	1.9235	5.0725	1.9893
95% CI of W_1	–	$[1.0397, 2.9496]$	$[3.8453, 6.4105]$	$[0.9899, 3.3032]$

Table 5.2: Force metrics for demonstrations, proposed method and baselines in the *Plant Insertion Task*. All statistics are computed using the equal-mass weighting scheme. Reported values include weighted mean, standard deviation, median, minimum force, and the Wasserstein-1 distance with 95% bootstrap confidence intervals. All force metrics are reported in Newtons (N).

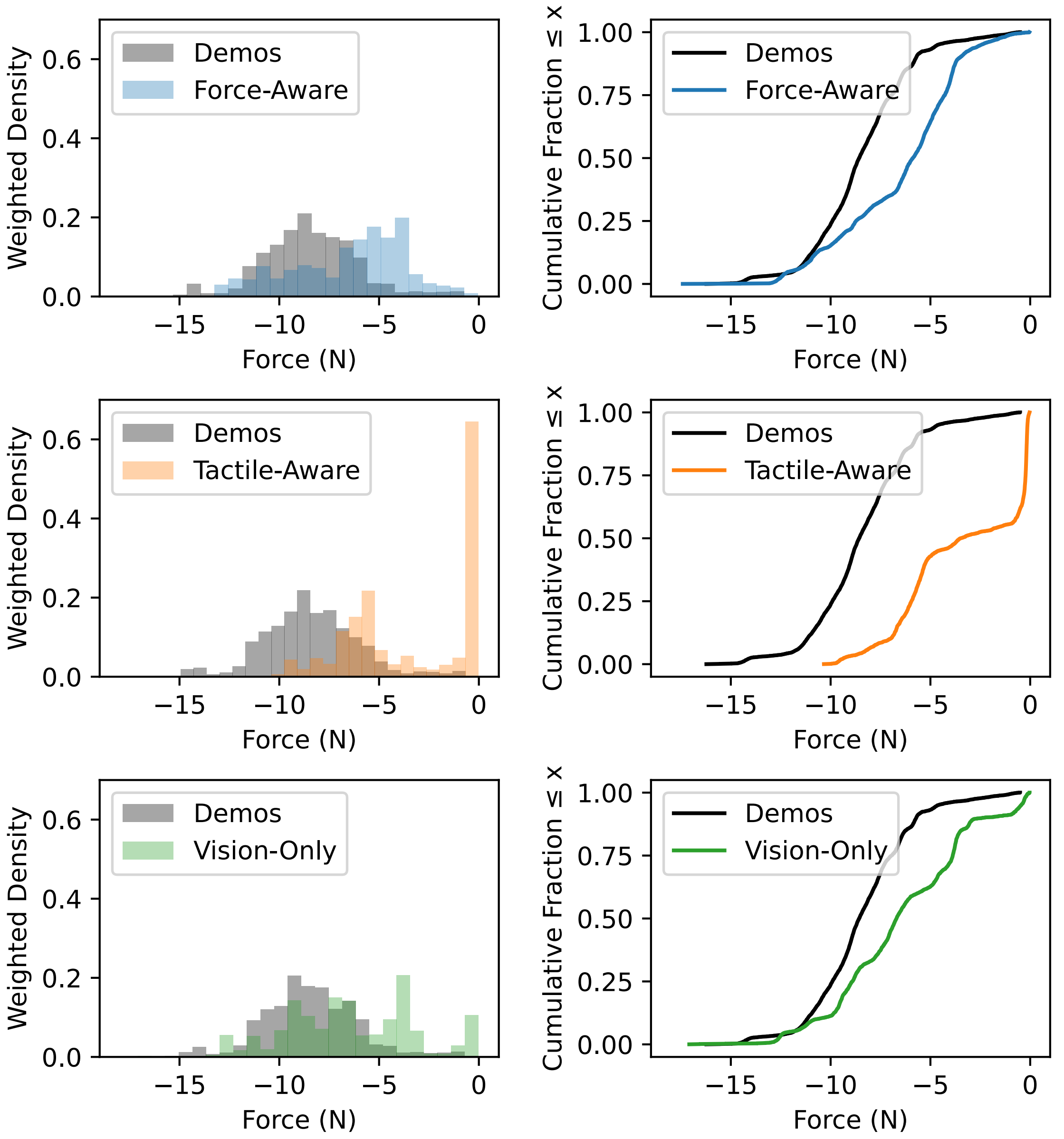


Figure 5.3: Weighted ECDFs and histograms of force values for demonstrations, proposed method, and baselines in the *Plant Insertion Task*. The ECDFs show the cumulative fraction of samples at or below each force value, where overlap means similar distributions and shifts indicate bias. The overlapping histograms highlight differences in force spread and magnitude between demonstrations and control strategies.

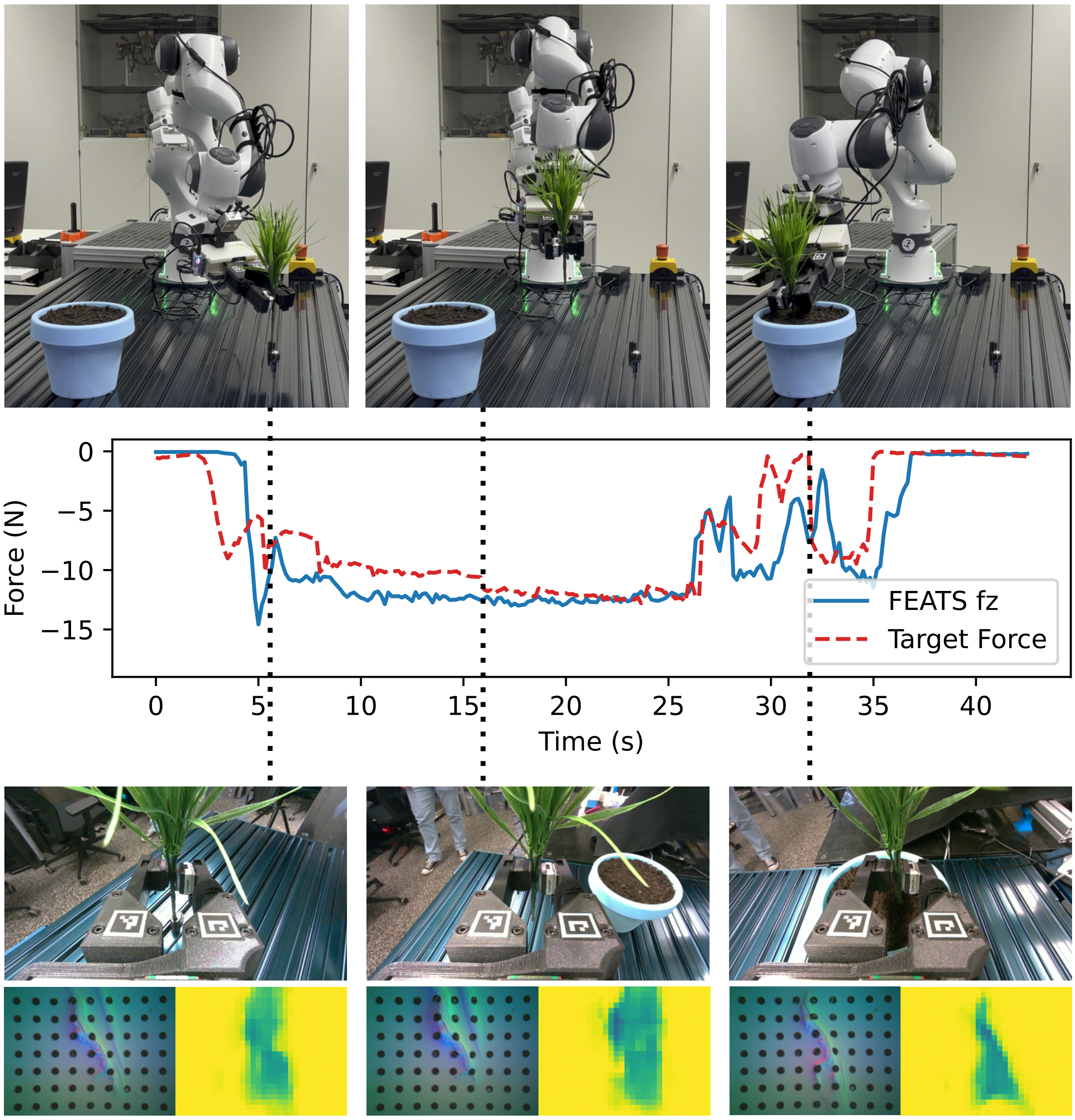


Figure 5.4: Illustration of a successful rollout with the *Force-Aware* strategy in the *Plant Insertion Task*. The plot shows the estimated normal force from FEATS and the target force predicted by the diffusion policy over time. Snapshots at three points along the trajectory include the external view of the robot, the image from the Intel RealSense D405 camera mounted on the actuated-UMI gripper, the GelSight Mini tactile image, and the corresponding FEATS normal force distribution estimation.

5.3 Grape Extraction Task

In the *Grape Extraction Task*, the robot must grasp a grape that is mounted on a toothpick, remove it without crushing or slipping, and place it into a bowl. The task demands delicate contact handling and minimal squeezing.

5.3.1 Experimental Setup

Fig. 5.5 shows the experimental setup used for the *Grape Extraction Task*. The setup again involves fixed object positions. The grape is manually put on a toothpick before each trial, and the target bowl remains in a constant location. As in the *Plant Insertion Task*, the robot executes a constrained initial motion before policy control begins. In this task, after grasp detection, the robot moves along the toothpick to detach the grape from the toothpick, after which full control is handed over to the policy.

Grasp detection and control logic followed the same procedure as in the *Plant Insertion Task*, with the same thresholds to trigger the transition from initial state motion to full policy control.

A particular challenge of this task is that grapes are slightly deformable, which makes force estimation with the GelSight Mini sensor more difficult. Since FEATS estimates forces based on gel deformation, the additional deformation of the grape introduces ambiguity. At higher applied forces, the grape deforms more significantly, which leads to less reliable force estimates. This poses a real problem for closed-loop force control.

A rollout is considered successful if the grape is placed in the bowl intact and visually undamaged, simulating a “salable” condition.

5.3.2 Experimental Results

In the *Grape Extraction Task*, across 20 rollouts, the *Force-Aware* strategy again achieves the highest success rate at 85% (cf. Fig. 5.6), reliably removing grapes from the toothpick without crushing them. A successful rollout using the *Force-Aware* strategy can be seen in Fig. 5.8. The *Tactile-Aware* strategy performs reasonably well with a success rate of 60%, but often fails when the grip is too weak and the grape slips out of the fingers. The

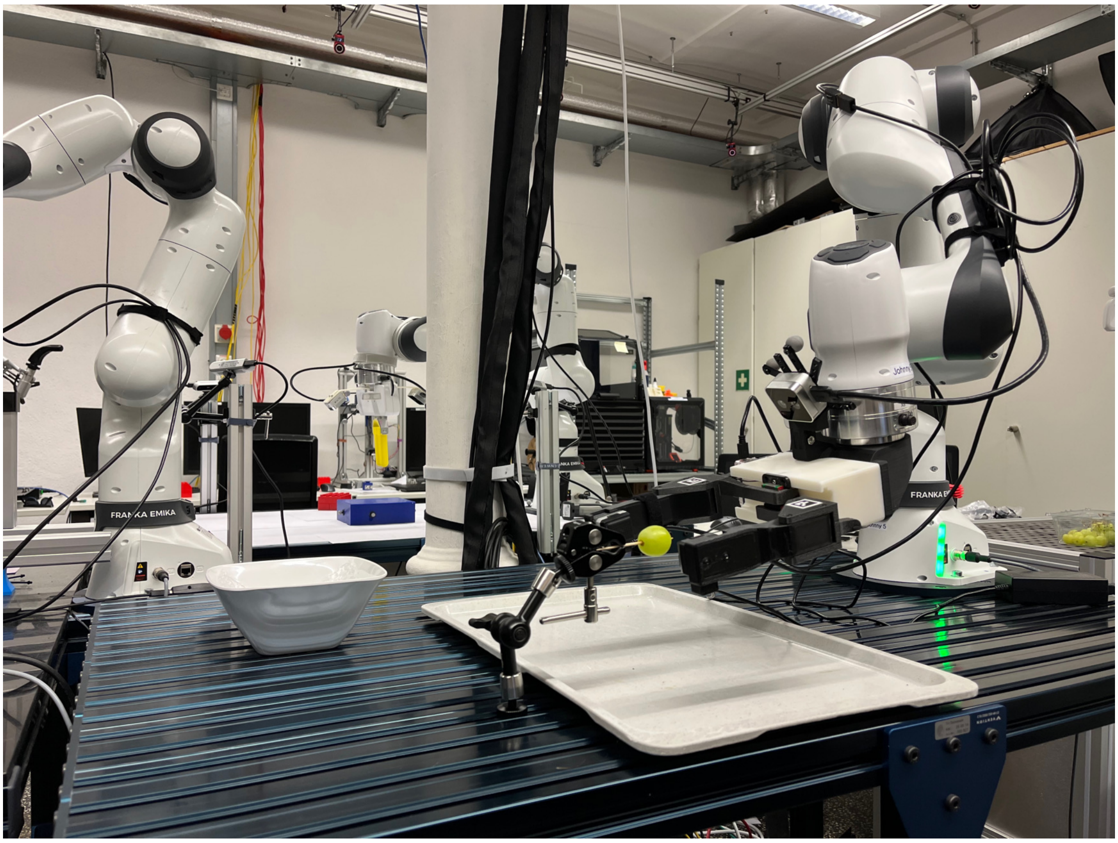


Figure 5.5: Experimental setup for the *Grape Extraction Task*. The image shows the Franka Research 3 robot equipped with the actuated-UMI gripper in its initial state before starting a rollout. The robot must remove the grape from a toothpick and place it in a bowl without damaging the grape.

Vision-Only strategy fails entirely, as its binary gripper command applies excessive forces, crushing the grapes.

Tab. 5.3 reports the weighted statistics and Wasserstein-1 distances. Demonstrations show a low-force profile with a mean of about $-3.6\,\mathrm{N}$ and a standard deviation of $1.4\,\mathrm{N}$. The *Force-Aware* strategy reproduces this distribution most closely, with a mean of $-2.1\,\mathrm{N}$ and a Wasserstein-1 distance of $1.56\,\mathrm{N}$ with a 95% CI of $[1.04\,\mathrm{N}, 2.28\,\mathrm{N}]$. The *Tactile-Aware* strategy lies in a similar range with a mean of $-1.99\,\mathrm{N}$ and a comparable Wasserstein-1 distance of $1.63\,\mathrm{N}$ with a 95% CI of $[0.95\,\mathrm{N}, 2.41\,\mathrm{N}]$. By contrast, the *Vision-Only* strategy is far from the demonstrations, with an extreme mean of $-25.4\,\mathrm{N}$ and a much larger Wasserstein-1 distance of $21.78\,\mathrm{N}$.

Fig. 5.7 shows the corresponding weighted ECDFs and histograms. The curves for the *Force-Aware* and *Tactile-Aware* strategies are both close to the demonstrations, though slightly shifted toward weaker forces, while the *Vision-Only* strategy has a strong shift toward higher forces.

These differences once again reflect the control principles of each strategy. The *Force-Aware* strategy benefits from its closed-loop force control, allowing it to apply just enough force to firmly grasp the grape without crushing it. The *Tactile-Aware* strategy lacks explicit force control but implicitly benefits from its tendency to apply smaller forces, which is safer in this low-force setting. However, it sometimes fails when insufficient grip force causes it to lose the grape. The *Vision-Only* strategy suffers from its coarse, binary gripper command. This results in the gripper clamping too hard without modulation, which leads to crushing the grapes every time.

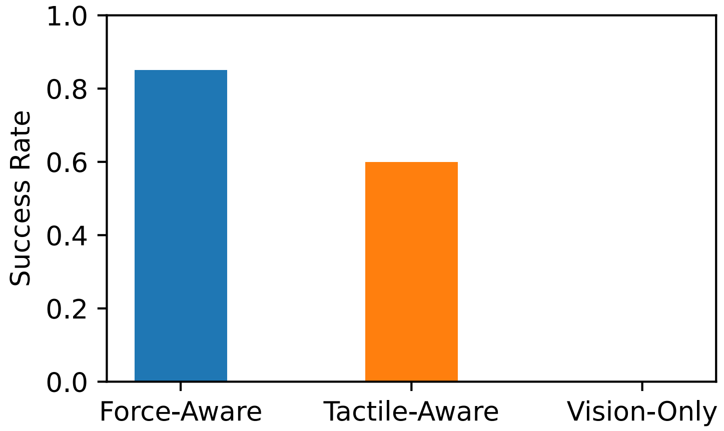


Figure 5.6: Success rates for the *Grape Extraction Task*. The *Force-Aware* strategy achieves the highest success rate at 85%, followed by the *Tactile-Aware* strategy at 60%. The *Vision-Only* strategy achieves 0% success and fails on all trials.

Metric (N)	Demo	Force-Aware	Tactile-Aware	Vision-Only
μ	-3.6209	-2.1277	-1.9932	-25.3917
σ	1.4344	1.8607	1.8443	9.9522
$\tilde{x}$	-3.5339	-1.8167	-1.7737	-26.2926
min	-12.0623	-13.6696	-7.6022	-51.5901
W_1	–	1.5634	1.6276	21.7809
95% CI of W_1	–	$[1.0361, 2.2843]$	$[0.9481, 2.4054]$	$[18.8638, 24.6033]$

Table 5.3: Force metrics for demonstrations, proposed method and baselines in the *Grape Extraction Task*. All statistics are computed using the equal-mass weighting scheme. Reported values include weighted mean, standard deviation, median, minimum force, and the Wasserstein-1 distance with 95% bootstrap confidence intervals. All force metrics are reported in Newtons (N).

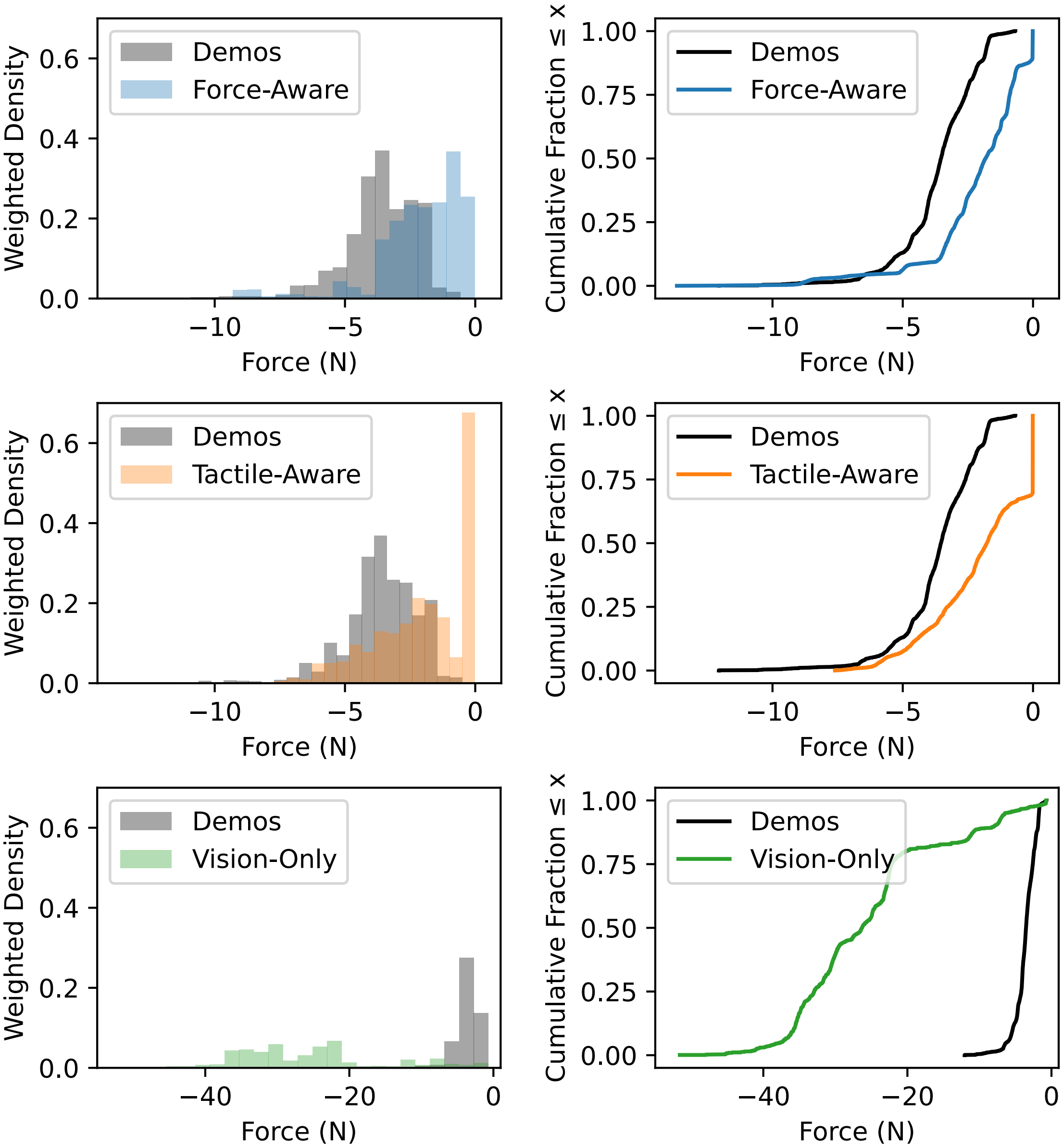


Figure 5.7: Weighted ECDFs and histograms of force values for demonstrations, proposed method, and baselines in the *Grape Extraction Task*. The ECDFs show the cumulative fraction of samples at or below each force value, where overlap means similar distributions and shifts indicate bias. The overlapping histograms highlight differences in force spread and magnitude between demonstrations and control strategies.

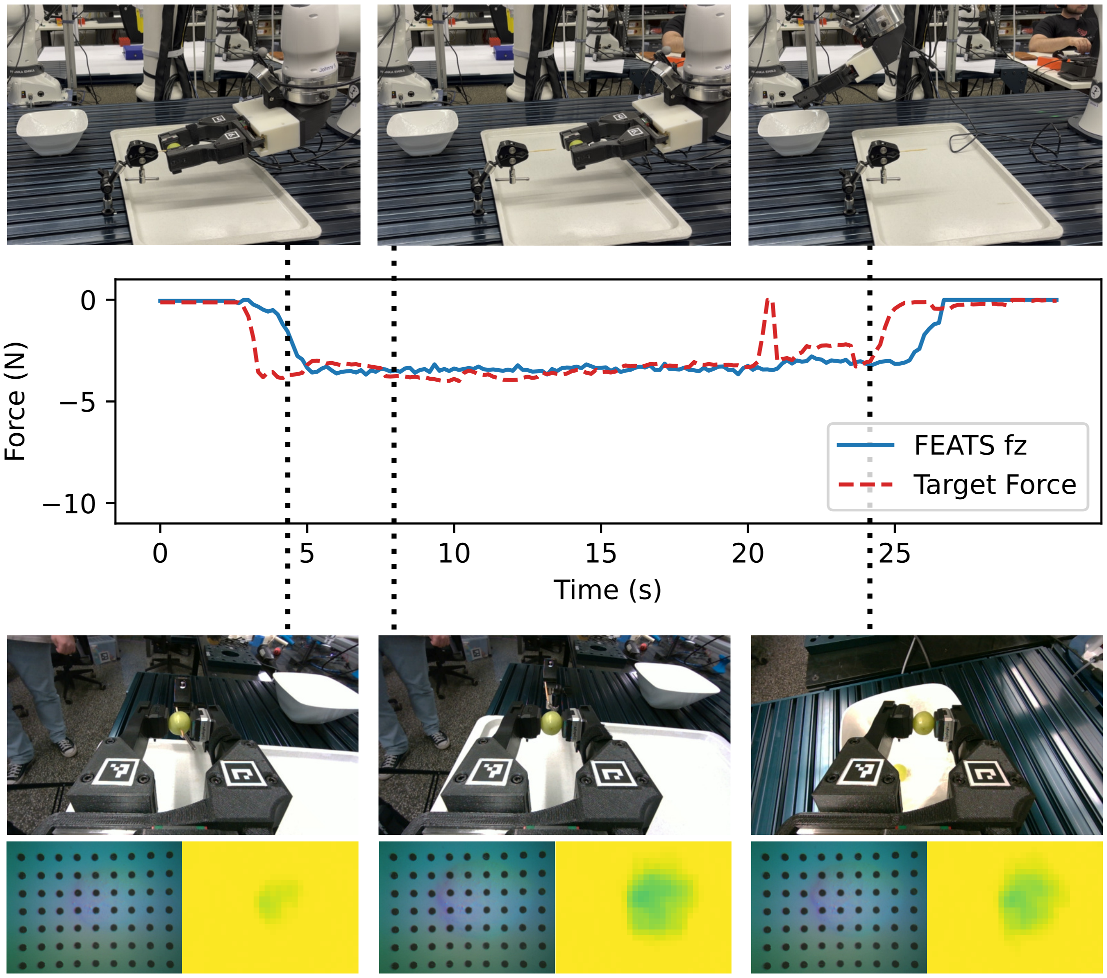


Figure 5.8: Illustration of a successful rollout with the *Force-Aware* strategy in the *Grape Extraction Task*. The plot shows the estimated normal force from FEATS and the target force predicted by the diffusion policy over time. Snapshots at three points along the trajectory include the external view of the robot, the image from the Intel RealSense D405 camera mounted on the actuated-UMI gripper, the GelSight Mini tactile image, and the corresponding FEATS normal force distribution estimation.

6 Discussion and Conclusion

This work investigated how tactile sensing can be incorporated not only as an observation modality but also directly into the action space of imitation learning for contact-rich manipulation. We introduced a *Force-Aware* diffusion policy that predicts target grip width and target grip force jointly with the robot pose. In this formulation, the force enters both the observation embedding of the policy and the action representation, producing temporally consistent, force-aware action sequences. Demonstrations were collected using an adapted hand-held UMI gripper equipped with a GelSight Mini sensor at one fingertip, with normal forces estimated via the FEATS model. To enable policy transfer, we developed the actuated-UMI gripper, whose geometry and kinematics match the hand-held UMI gripper. We executed gripper actions through a dual-mode controller that switches between grip width position control and closed-loop force control, depending on whether the measured and predicted forces exceed a threshold.

We evaluated our approach on two real-world tasks that spanned opposite ends of the force spectrum: a *Plant Insertion Task* requiring high-force contact to push a plastic plant into real soil, and a *Grape Extraction Task* demanding delicate, low-force manipulation to remove a grape from a toothpick without crushing it. Across these contrasting force settings, the *Force-Aware* policy consistently outperformed baselines. Explicit, closed-loop force control both improved task success and yielded applied forces closer to those demonstrated by an expert. In comparison, the *Vision-Only* baseline was brittle. Its binarized gripper control was sufficient when higher forces were preferable, but it failed catastrophically when excessive force caused breakage. The *Tactile-Aware* baseline, which relied on tactile sensing solely for observation, was able to detect contact but, without an explicit action-level force target, it frequently applied insufficient grip forces that resulted in failures.

Taken together, the results show that incorporating force directly into the action space and closing the loop on it can substantially improve imitation learning policies for contact-rich manipulation, enabling them to handle both delicate and high-force interactions with greater reliability. This work argues for *Force-Aware* imitation learning as a strong default for contact-rich manipulation, showing that imitation learning for manipulation tasks does not have to stop at kinematics. Adding force as an explicit action results in more reliable and safer performance across diverse task demands. Our approach further illustrates how high-level force targets can be integrated into simple dual-control schemes, providing a practical way to use force alongside grip width as a target. Future work should investigate how to incorporate richer tactile feedback into this framework, moving beyond a single normal force value to include full normal and shear force distributions. Additionally, replacing the hand-tuned contact threshold with an end-to-end learned contact-state estimator could lead to more reliable switching between control modes. Finally, a broader evaluation across a more diverse set of tasks would provide a stronger test of generalization and adaptability.

Bibliography

This thesis was independently written and linguistically revised with the assistance of DeepL and ChatGPT.

[1] Trevor Ablett et al. "Multimodal and Force-Matched Imitation Learning With a See-Through Visuotactile Sensor". In: *IEEE Transactions on Robotics* 41 (2025), pp. 946–959. ISSN: 1941-0468. DOI: 10.1109/tro.2024.3521864. URL: http://dx.doi.org/10.1109/TRO.2024.3521864.

[2] Ademi Adeniji et al. *Feel the Force: Contact-Driven Learning from Humans*. 2025. arXiv: 2506.01944 [cs.RO]. URL: https://arxiv.org/abs/2506.01944.

[3] Michael Bain and Claude Sammut. "A Framework for Behavioural Cloning." In: *Machine intelligence 15*. 1995, pp. 103–129.

[4] Raunaq Bhirangi et al. *AnySkin: Plug-and-play Skin Sensing for Robotic Touch*. 2024. arXiv: 2409.08276 [cs.RO]. URL: https://arxiv.org/abs/2409.08276.

[5] Antonio Bicchi and Vijay Kumar. "Robotic grasping and contact: A review". In: *Proceedings 2000 ICRA. Millennium conference. IEEE international conference on robotics and automation. Symposia proceedings (Cat. No. 00CH37065)*. Vol. 1. IEEE. 2000, pp. 348–353.

[6] Aude Billard and Danica Kragic. "Trends and challenges in robot manipulation". In: *Science* 364.6446 (2019), eaat8414.

[7] Alina Böhm et al. "What matters for active texture recognition with vision-based tactile sensors". In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2024, pp. 15099–15105.

[8] Claire Chen et al. *DexForce: Extracting Force-informed Actions from Kinesthetic Demonstrations for Dexterous Manipulation*. 2025. arXiv: 2501.10356 [cs.RO]. URL: https://arxiv.org/abs/2501.10356.

[9] Wei Chen et al. “Tactile sensors for friction estimation and incipient slip detection—Toward dexterous robotic manipulation: A review”. In: *IEEE Sensors Journal* 18.22 (2018), pp. 9049–9064.

[10] Cheng Chi et al. “Diffusion Policy: Visuomotor Policy Learning via Action Diffusion”. In: *The International Journal of Robotics Research* (2024).

[11] Cheng Chi et al. “Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots”. In: *arXiv preprint arXiv:2402.10329* (2024).

[12] The SciPy community. *scipy.stats.wasserstein_distance*. https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.wasserstein_distance.html. Accessed: 2025-08-14.

[13] Mark R Cutkosky and James M Hyde. “Manipulation control with dynamic tactile sensing”. In: *6th international symposium on robotics research, Hidden Valley, Pennsylvania*. 1993.

[14] Ravinder S. Dahiya et al. “Tactile Sensing—From Humans to Humanoids”. In: *IEEE Transactions on Robotics* 26.1 (2010), pp. 1–20. DOI: 10.1109/TRO.2009.2033627.

[15] Niklas Funk et al. *Evetac: An Event-based Optical Tactile Sensor for Robotic Manipulation*. 2024. arXiv: 2312.01236 [cs.RO]. URL: https://arxiv.org/abs/2312.01236.

[16] Niklas Funk et al. “High-resolution pixelwise contact area and normal force estimation for the gelsight mini visuotactile sensor using neural networks”. In: *Embracing Contacts-Workshop at ICRA 2023*. 2023.

[17] Niklas Funk et al. *On the Importance of Tactile Sensing for Imitation Learning: A Case Study on Robotic Match Lighting*. 2025. arXiv: 2504.13618 [cs.RO]. URL: https://arxiv.org/abs/2504.13618.

[18] GelSight, Inc. *GelSight Mini - Datasheet*. https://www.gelsight.com/wp-content/uploads/2023/01/GelSight_Datasheet_GSMini_12.20.22.pdf. Accessed: 2025-06-08.

[19] Ningquan Gu, Kazuhiro Kosuge, and Mitsuhiro Hayashibe. “TactileAloha: Learning Bimanual Manipulation With Tactile Sensing”. In: *IEEE Robotics and Automation Letters* 10.8 (2025), pp. 8348–8355. DOI: 10.1109/LRA.2025.3585396.

[20] Yunhai Han et al. *Learning Generalizable Vision-Tactile Robotic Grasping Strategy for Deformable Objects via Transformer*. 2023. arXiv: 2112.06374 [cs.RO]. URL: https://arxiv.org/abs/2112.06374.

[21] Erik Helmut et al. *Learning Force Distribution Estimation for the GelSight Mini Optical Tactile Sensor Based on Finite Element Analysis*. 2025. arXiv: 2411.03315 [cs.RO]. URL: https://arxiv.org/abs/2411.03315.

[22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. *Denoising Diffusion Probabilistic Models*. 2020. arXiv: 2006.11239 [cs.LG]. URL: https://arxiv.org/abs/2006.11239.

[23] Binghao Huang et al. "3D ViTac:Learning Fine-Grained Manipulation with Visuo-Tactile Sensing". In: *Proceedings of Robotics: Conference on Robot Learning(CoRL)*. 2024.

[24] Michael Janner et al. "Planning with diffusion for flexible behavior synthesis". In: *arXiv preprint arXiv:2205.09991* (2022).

[25] Shulong Jiang et al. *GelFusion: Enhancing Robotic Manipulation under Visual Constraints via Visuotactile Fusion*. 2025. arXiv: 2505.07455 [cs.RO]. URL: https://arxiv.org/abs/2505.07455.

[26] Roland S Johansson and Göran Westling. "Signals in tactile afferents from the fingers eliciting adaptive motor responses during precision grip". In: *Experimental brain research* 66.1 (1987), pp. 141–154.

[27] Roland S Johansson and Goran Westling. "Roles of glabrous skin receptors and sensorimotor memory in automatic control of precision grip when lifting rougher or more slippery objects". In: *Experimental brain research* 56.3 (1984), pp. 550–564.

[28] Mohsen Kaboli, Kunpeng Yao, and Gordon Cheng. "Tactile-based manipulation of deformable objects with dynamic center of mass". In: *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*. IEEE. 2016, pp. 752–757.

[29] Mike Lambeta et al. "Digit: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation". In: *IEEE Robotics and Automation Letters* 5.3 (2020), pp. 3838–3845.

[30] Mike Lambeta et al. *Digitizing Touch with an Artificial Multimodal Fingertip*. 2024. arXiv: 2411.02479 [cs.RO]. URL: https://arxiv.org/abs/2411.02479.

[31] Nathan F Lepora. "Soft biomimetic optical tactile sensing with the TacTip: A review". In: *IEEE Sensors Journal* 21.19 (2021), pp. 21131–21143.

[32] Fangchen Liu et al. *ViTaMIn: Learning Contact-Rich Tasks Through Robot-Free Visuo-Tactile Manipulation Interface*. 2025. arXiv: 2504.06156 [cs.RO]. URL: https://arxiv.org/abs/2504.06156.

[33] Willow Mandil et al. “Tactile-sensing technologies: Trends, challenges and outlook in agri-food manipulation”. In: *Sensors* 23.17 (2023), p. 7362.

[34] Louise R Manfredi et al. “Natural scenes in tactile texture”. In: *Journal of neurophysiology* 111.9 (2014), pp. 1792–1802.

[35] Xiaofeng Mao et al. *Learning Fine Pinch-Grasp Skills using Tactile Sensing from A Few Real-world Demonstrations*. 2024. arXiv: 2307.04619 [cs.RO]. URL: https://arxiv.org/abs/2307.04619.

[36] D.Q. Mayne and H. Michalska. “Receding horizon control of nonlinear systems”. In: *Proceedings of the 27th IEEE Conference on Decision and Control*. 1988, 464–465 vol.1. DOI: 10.1109/CDC.1988.194354.

[37] Andrew Y Ng, Stuart Russell, et al. “Algorithms for inverse reinforcement learning.” In: *Icml*. Vol. 1. 2. 2000, p. 2.

[38] Takayuki Osa et al. “An algorithmic perspective on imitation learning”. In: *Foundations and Trends® in Robotics* 7.1-2 (2018), pp. 1–179.

[39] Ethan Perez et al. *FiLM: Visual Reasoning with a General Conditioning Layer*. 2017. arXiv: 1709.07871 [cs.CV]. URL: https://arxiv.org/abs/1709.07871.

[40] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. May 18, 2015. DOI: 10.48550/arXiv.1505.04597. arXiv: 1505.04597[cs].

[41] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[42] Carmelo Sferrazza et al. “Ground truth force distribution for learning-based tactile sensing: A finite element approach”. In: *IEEE Access* 7 (2019), pp. 173438–173449.

[43] Sagar Sharma, Yonghyun Kim, and Chung Hyuk Park. “Visuotactile Diffusion Policy: Automated Failure Recovery in Assistive Tasks with Tactile Manipulation Using Imitation Learning”. In: *2025 22nd International Conference on Ubiquitous Robots (UR)*. 2025, pp. 294–300. DOI: 10.1109/UR65550.2025.11077988.

[44] Roger Y Tsai, Reimar K Lenz, et al. “A new technique for fully autonomous and efficient 3 d robotics hand/eye calibration”. In: *IEEE Transactions on robotics and automation* 5.3 (1989), pp. 345–358.

[45] Gary R VandenBos. *APA dictionary of psychology.* American Psychological Association, 2007.

[46] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

[47] Nicholas Wettels et al. “Multi-modal synergistic tactile sensing”. In: *Tactile sensing in humanoids—Tactile sensors and beyond workshop, 9th IEEE-RAS international conference on humanoid robots*. 2009.

[48] Longyan Wu et al. *FreeTacMan: Robot-free Visuo-Tactile Data Collection System for Contact-rich Manipulation*. 2025. arXiv: `2506.01941 [cs.RO]`. URL: `https://arxiv.org/abs/2506.01941`.

[49] Han Xue et al. *Reactive Diffusion Policy: Slow-Fast Visual-Tactile Policy Learning for Contact-Rich Manipulation*. 2025. arXiv: `2503.02881 [cs.RO]`. URL: `https://arxiv.org/abs/2503.02881`.

[50] Kelin Yu et al. *MimicTouch: Leveraging Multi-modal Human Tactile Demonstrations for Contact-rich Manipulation*. 2025. arXiv: `2310.16917 [cs.RO]`. URL: `https://arxiv.org/abs/2310.16917`.

[51] Wenzhen Yuan, Siyuan Dong, and Edward H Adelson. “Gelsight: High-resolution robot tactile sensors for estimating geometry and force”. In: *Sensors* 17.12 (2017), p. 2762.

[52] Wenzhen Yuan et al. “Measurement of shear and slip with a GelSight tactile sensor”. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2015, pp. 304–311.

[53] Yi Zhou et al. “On the Continuity of Rotation Representations in Neural Networks”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2019.

[54] Yuan Zhu et al. “Recent advances in resistive sensor technology for tactile perception: A review”. In: *IEEE sensors journal* 22.16 (2022), pp. 15635–15649.