Teleoperation-Driven and Keyframe-Based Generalizable Imitation Learning for Construction Robots

Yan Li[1], Songyang Liu[2], Mengjun Wang[3], Shuai Li[4], Jindong Tan[5]

1Postdoctoral Researcher, Department of Civil and Environmental Engineering, The University of Tennessee, Knoxville, 37996; E-mail: yli141@vols.utk.edu

2PhD. Candidate, Department of Civil and Environmental Engineering, The University of Tennessee, Knoxville, 37996; Email: sliu78@vols.utk.edu

3PhD. Candidate, Department of Civil and Environmental Engineering, The University of Tennessee, Knoxville, 37996; E-mail: mwang43@vols.utk.edu

4Associate Professor, PhD, Department of Civil and Environmental Engineering (corresponding author), The University of Tennessee, Knoxville, 37996; Email: sli48@utk.edu

5Professor, PhD, Mechanical, Aerospace and Biomedical Engineering, The University of Tennessee, Knoxville, 37996; Email: tan@utk.edu

Abstract

The construction industry has long been plagued by low productivity and high injury and fatality rates. Robots have been envisioned to automate the construction process, thereby substantially improving construction productivity and safety. Despite the enormous potential, teaching robots to perform complex construction tasks is challenging. We present a generalizable framework to harness human teleoperation data to train construction robots to perform repetitive construction tasks. First, we develop a teleoperation method and interface to control robots on construction sites, serving as an intermediate solution toward full automation. Teleoperation data from human operators, along with context information from the job site, can be collected for robot learning. Second, we propose a new method for extracting keyframes from human operation data to reduce noise and redundancy in the training data, thereby improving robot learning

efficacy. We propose a hierarchical imitation learning method that incorporates the keyframes to train the robot to generate appropriate trajectories for construction tasks. Third, we model the robot's visual observations of the working space in a compact latent space to improve learning performance and reduce computational load. To validate the proposed framework, we conduct experiments teaching a robot to generate appropriate trajectories for excavation tasks from human operators' teleoperations. The results suggest that the proposed method outperforms state-of-the-art approaches, demonstrating its significant potential for application.

Keywords

Construction Robots; Trajectory Generation; Imitation Learning; Keyframe; Teleoperation

Introduction

The construction industry, valued at $13 trillion in 2021 and growing at a compound annual growth rate (CAGR) of 9.8% to reach more than 23 trillion by 2026, is an essential component of the global economy (Andrew Reynolds 2022). However, the construction industry faces long-standing issues, including an aging workforce, as well as safety and health problems. Construction work, which is dangerous, physically demanding, and cognitively challenging, has traditionally been performed by an aging and diverse workforce in unstructured and dynamic environments. The low productivity results in 98% of projects having cost overruns and 77% suffering from schedule delays (Sriram et al. 2015). The construction industry has the highest number of fatalities and the highest rate of work-related musculoskeletal disorders (Wang et al. 2017). Out of the 4,779 worker fatalities in private industry in 2018, 1,008 or 21.1% were in construction (Occupational Safety and Health Administration 2018). In addition, 44.6% of all construction injuries and illnesses were related to musculoskeletal disorders, and the lifetime risk of overexertion injuries in construction is 21% (The National Institute for Occupational Safety and Health (NIOSH) 2019). There is great but unconsolidated potential for robotic construction to improve work productivity, safety, and workers' occupational health (Saidi et al. 2016).

Robots are envisioned for deployment on construction sites to assist with physically demanding work, relieve workers from repetitious tasks, and protect them from on-site risks. Despite their great potential, the challenge of imbuing robots with the intelligence to navigate the unstructured and dynamic environments of construction sites, and execute complex tasks, remains formidable. Teleoperation-based methods, while allowing for direct control of robots, introduce significant challenges, including the need for extensive training that imposes high costs in both time and resources, compounded by the scarcity of skilled workers. Learning-based methods emerge as a solution, enabling robots to learn from data, improve over time, and adapt to diverse tasks and environments without heavy reliance on skilled operators. This approach not only addresses the limitations of teleoperation but also capitalizes on its immediate benefits, offering a balanced path forward in the dynamic construction environment. Developing methods to teaching robots without relying on expert intervention stands out as a significant step in advancing their operational capabilities within construction environments. At present, robot operations in these environments rely heavily on the expertise of skilled professionals, through either manual programming or data-driven approaches. Our research aims to democratize this process by enabling workers with varying levels of robotics proficiency to transfer their job skills to robots, thus facilitating automated task execution. By moving away from exclusive reliance on experts, we aim to foster a more inclusive and accessible approach to deploying robots in construction contexts.

Efficiently teaching robots presents inherent challenges, complicated by the balance between data scarcity and the necessity for effective demonstrations. To overcome this hurdle, we propose an integrated methodology that merges mimic learning with reinforcement learning. Our approach adeptly handles both task acquisition and execution exploration, thus eliminating the need for expert involvement in the teaching process. Aligned with human learning principles, our study involves two main components: first, evaluating the teacher's experience and extracting key points; and second, learning from these key points and the teacher's experience. The first part involves evaluating the teacher's experience and extracting key points from trajectory data. We propose a keyframe identification method to reduce data volume and dimensions

for training. The second part comprises an imitation learning framework combined with a goal-conditioned reinforcement learning model, enabling robots to learn from human demonstrations while maintaining their exploration capabilities. Additionally, we transform the data into a low-dimensional latent space representation to facilitate learning process. We propose a hierarchical reinforcement learning structure and a generative-adversarial-like keyframe classification structure. The hierarchical structure includes a subgoal generation network and a primitive motion network. Moreover, we propose a vision-based trajectory generation method that leverages latent space exploration to reduce computation load and enhance learning performance.

The rest of the paper is organized as follows. Section 2 reviews relevant literature in construction robotics and robot learning methods, Section 3 illustrates the framework and methods, including keyframe extraction, robot learning framework and the latent space generation, Section 4 presents the experiment results and compares the performance with the state-of-the-art, Section 5 discusses the applicability of the proposed method and its limitations, Section 6 concludes the paper with remarks on the contributions to knowledge and insights for practical application.

Literature review

Pre-programmed and Teleoperated Robotics

The development of construction robotics is changing the management and execution of construction projects. There are three types of construction robots: pre-programmed, teleoperated, and learning-based robots (Saidi et al. 2016). Pre-programmed construction robots are highly automated and can be used for various construction tasks without human intervention. (Gambao et al. 2000) designed an integrated automated robotic system to handle the shuttering and installation of plane-parallel blocks during the assembly of building blocks. (Yu et al. 2009) integrated a pattern generation algorithm into an automated brick-laying system to perform brick handling on construction sites. (Keating and Oxman 2013) designed a multi-functional, pre-programmed robotic arm platform. This platform utilizes major manufacturing

technologies including additive, formative and subtractive fabrication. (Lublasser et al. 2018) proposed a robot-based method to apply formwork concrete onto the bare walls of existing buildings. This method provides a facade finish that is insulating and recyclable, where the motions of the robotic arm are programmed. These pre-programmed construction robots have the potential to save both time and money. However, they can only carry out the duties for which they were designed and are unable to adapt automatically to changes in construction sites. Teleoperated construction robots lower the risk of accidents and injuries by enabling human workers to complete tasks from a secure location. Also, they are adaptable and capable of doing various tasks, such as demolition, excavation, and material handling. Teleoperated humanoid robots have been well-developed to remotely operate various industrial vehicles like lift trucks, and backhoes at construction sites (Hasunuma et al. 2002, 2003; Yokoi et al. 2006). (Kim et al. 2009) designed a teleoperated excavator system to help avoid workers' risks while operating the excavator on inclined planes. Control data for this system is captured from sensors attached to the operators' arms and then transmitted via Bluetooth. (David et al. 2014) designed a system merging information from real and virtual worlds to help workers remotely perform inspection and maintenance of on-site tunnel boring machine. (Liu et al. 2021) and (Xia et al. 2023) developed a remote-control system that converts signals received from a wearable electroencephalogram device into commands for robots. This innovative approach enhances workers' control in environments such as underwater and space construction, where the workers' ability to manually steer the robots is constrained. However, the latency between the operator's commands and the construction robot's actions has been a common issue (Falanga et al. 2019; Luck et al. 2006). Meanwhile, the teleoperated robotics still need human intervention to monitor their status and issue commands in real time. To address these issues, researchers have investigated learning-based robotics which can automatically complete the job while adapting to changes in construction sites.

Learning-based Robotics

Construction robots using reinforcement learning (RL) technology are autonomous machines that may gain knowledge from their mistakes and hone their accuracy and productivity over time. Many RL

methods have been developed for applying learning-based robotics on construction sites, enhancing efficiency and safety. (Apolinarska et al. 2021) applied an adapted Deep Deterministic Policy Gradient algorithm (DDPG) (Lillicrap et al. 2016) algorithm to train robots for assembling lap joints in custom timber frames as inserting a timber element into its mating counterpart(s). (Belousov et al. 2022) proposed a Twin Delayed DDPG (TD3) (Fujimoto et al. 2018) based RL method to train robots to assemble a structure from predefined discrete building blocks autonomously, like stacking blocks on placed blocks. (Lee and Kim 2021) developed an automated construction hoist trained by deep Q-network (DQN) (Mnih et al. 2013) to reduce the number of unnecessary trips when performing lift tasks. These studies mainly focus on training robots from scratch which could reduce learning efficiency.

Utilizing expert demonstrations to train RL-based robots has been well investigated to reduce unnecessary explorations and improve learning efficiency (Fang et al. 2019; Li and Zou 2023; Pfeiffer et al. 2018; Pore et al. 2021; Zhou et al. 2023b; a). (Huang et al. 2023) trained RL-based construction robots to learn long-horizon tasks like picking and installing window panels from demonstrations in virtual reality (VR). (Duan and Zou 2023) collected intuitive expert demonstration using VR platform where a robot will automatically follow the position, rotation, and actions of the expert's hand in real-time, instead of requiring an expert to control the robot via controllers. However, controlling real construction robots and their perception of the environment are significantly more complex than what is simulated. Many methods have been proposed to adapt the trained RL policy directly to real construction robot. (Liu et al. 2018) developed a framework for robot learning to imitate behaviors from expert demonstration videos. (Liang et al. 2019, 2020) proposed a Learning from Demonstration (LfD) method to teach robots to perform quasi-repetitive construction tasks like installing ceiling tiles from expert demonstration videos. For this training approach to yield strong model performance, a substantial number of high-dimensional videos is typically required. As previously indicated, expert demonstrations may contain a significant amount of redundant data. To solve this issue, we propose a keyframe-based learning system. Researchers have employed keyframe extraction methods to improve computational efficiency for robot learning (Hartmann et al. 2021; Zhao and

Cheah 2023). While (Hartmann et al. 2021) enhances task division and automation for scalability in multi-robot systems, our contribution highlights the importance of teleoperation in bridging human expertise with robotic capabilities, particularly in unpredictable or intricate construction environments. Unlike (Zhao and Cheah 2023), which relies on an automated BIM-based system using object detection for robot initialization, our work extends the capability of robots to learn from their operations and environments over time.

Reinforcement Learning in Robotics: Developing robots to accomplish tasks has been extensively explored, utilizing both model-based and model-free approaches. Model-based methods, referenced in (Abdolmaleki et al. 2018; Song et al. 2019; Zakka et al. 2019; Zeng et al. 2020), often incorporate human-defined primitives to guide robot actions. Although effective in certain scenarios, these methods struggle with generalization across diverse task types due to the vast range of potential primitives. Conversely, model-free approaches (Ding et al. 2019; Haarnoja et al. 2018; Ho and Ermon 2016; Nasiriany et al. 2019; Zhu et al. 2020) offer flexibility but face challenges such as high variance in pose estimation and prolonged training times due to reward sparsity. Both paradigms aim to address the complex requirements of long-horizon and temporally extended tasks, with strategies including compositional policy structures derived from demonstrations (Abdolmaleki et al. 2018), manually specified primitives (Kabir et al. 2020), learned temporal abstractions (Chane-Sane et al. 2021), and direct model-free reinforcement learning (Schulman et al. 2017).

Challenges in Imitation Learning: Within the imitation learning (IL) framework, both behavior cloning and inverse reinforcement learning face distinct challenges. Behavior cloning methods are known for their substantial data requirements and the propensity to inherit bias from the training dataset. Inverse RL, while powerful for deriving reward functions from observed behaviors, often struggles with learning comprehensive reward functions that encompass entire trajectories. These methods have yet to overcome the limitation of requiring successful examples to facilitate model training effectively.

Addressing the Limitations: Our method introduces a novel approach to surmounting these hurdles. By integrating a new experience relabeling method and an action evaluation network, we directly address the issues of off-trajectory actions and the reward sparsity common in complex visual manipulation tasks. This innovation allows for more precise bottleneck estimation and alleviates the high variance issue associated with model-free paradigms. Furthermore, our approach mitigates the restrictive assumptions classical planning methods make about state space and state connectivity, enhancing applicability to a broader range of complex tasks.

Developing robots to accomplish tasks has been a well-studied problem (Hentout et al. 2019; Jing et al. 2018), these methods can be categorized into model-based and model-free approaches. Existing model-based methods typically do not perform well for indefinite problems. Behavior cloning methods are usually data-ravenous, and the results are biased (Codevilla et al. 2019). Inverse RL methods are hard to learn reward function for whole trajectory (Arora and Doshi 2018). Moreover, the model-free paradigm such as (Ho and Ermon 2016; Nasiriany et al. 2019) suffers from high variance in pose estimation, resulting in a lack of precision in bottleneck estimation. Model-based approaches (Abdolmaleki et al. 2018; Song et al. 2019; Zakka et al. 2019; Zeng et al. 2020) usually come with human-introduced primitives and train the robot actions based on these primitives. However, in trajectory generation tasks, the primitives vary in an enormous range, and they are hard to generalize to other types of tasks. Model-free methods (Ding et al. 2019; Haarnoja et al. 2018; Ho and Ermon 2016; Nasiriany et al. 2019; Zhu et al. 2020) typically take longer training times and suffer from reward sparsity. The long horizon and the temporally extended tasks enable the robot to perform a diverse set of tasks (Finn et al. 2015; Jayaraman et al. 2018; Thakar et al. 2018). These approaches have added compositional structure to policies, either from demonstration (Abdolmaleki et al. 2018), with manually specified primitives (Kabir et al. 2020), learned temporal abstractions (Chane-Sane et al. 2021), or through model-free RL (Schulman et al. 2017). These works have studied such hierarchy in grid worlds and simulated control tasks with known reward functions. Classical planning methods have proven effective in performing long-horizon tasks. However, they make restrictive

assumptions about the state space and the connectivity between states. This limits their applicability to complex visual manipulation tasks. In these methods, the problem of the off-trajectory actions and the reward sparsity of complex tasks are not solved. In our method, we propose a novel experience relabeling method and an action evaluation network to address these two problems.

Vision-based Trajectory Generation with Latent Space Exploration

The feasibility of keyframe-guided trajectory generation in learning-based robotics was discussed in the Introduction chapter. However, in practical applications, obtaining accurate object and target information is challenging without supplementary algorithms. Utilizing various sensors, such as laser, radar, and RGB-D camera, has been well investigated to solve this problem. (Mandlekar et al. 2019) used a crowd-sourced dataset that has image observations from a front view camera to let robotic arm learn control policies. (Praveena et al. 2019) proposed a handheld grabber tool, equipped with force-torque sensor, providing accurate measurement of the applied forces and torques when grasping objects. (Zeng et al. 2022) developed educational robots sharing certain characteristics including the focus on assistive functions like buttons, grayscale sensors, and cameras. However, these methods do not solve two main problems: First, the computation load of the reinforcement learning model is massive, resulting in difficulty and latency in the training process. To tackle this problem, our method models the vision space into the latent space. Second, RL methods typically perform poorly with high dimensional inputs. To tackle this problem, a trendy way to deal with image information is to use an encoder-decoder network to reduce the reinforcement learning network input dimension. (Abdolmaleki et al. 2018) argued that existing reinforcement learning algorithms can be expensive in terms of sample requirements and suffer from high gradient variance, resulting in unstable learning and slow convergence. To keep track of reachable latent states, (Bharadhwaj et al. 2020) proposed a distance-conditioned reachability network that is trained to infer whether one state is reachable from another within the specified latent space distance. A conventional algorithm comprises an image depth information encoder and a reinforcement learning framework. In this paper, we also adopt this framework approach.

One limitation of existing vision-based robot manipulation learning methods is the requirement for a carefully constructed environment. Repeatedly setting up the environment can introduce configuration errors. In the study by (Luo et al. 2021), the task involves robot manipulation using a monocular vision system. The critical aspect of environment setup is the camera's installation position, which affects the ability to obtain spatial information about the target from a single camera. Incorrect camera positioning can lead to issues with occlusion. In the study by (Zhou et al. 2022), the task involves robotic fruit grasping under leaf interference. The positioning of the leaves and fruit is crucial because it affects the robot's ability to grasp the fruit. In the study by (Liu et al. 2020), the robot learns policy autonomously by interacting with the environment. The setup of the robot and object states is important because it affects learning efficiency. Therefore, careful environment setup is critical to ensuring the accuracy and efficiency of robot manipulation tasks. To tackle this problem, we propose a robot end-effector position estimation network to match the robot encoder position with the estimated position.

The goal of learning from vision-based demonstrations is to map the image with the target goal states (Song et al. 2019; Zeng et al. 2020). A key limitation of many existing methods is their requirement for predefined goal states and positions. (Lenz et al. 2013) presented a two-step cascaded system with two deep networks for detecting robotic grasps in an RGB-D view of a scene, effectively avoiding the need for time-consuming hand-design of features. (Hester et al. 2017a) introduced a novel algorithm that utilizes small sets of demonstration data to significantly accelerate the learning process in deep RL, addressing the issue of RL algorithms typically requiring large amounts of data before achieving reasonable performance. (Zhang et al. 2020) proposed a hierarchical path planning framework, SG-RL, which combines geometric path-planning with RL to plan rational paths in continuous and uncertain environments. (Kabir et al. 2019) presented a non-linear optimization problem for path-constrained trajectory generation in multi-robot systems. (Hester et al. 2017b) presented an algorithm called Deep Q-learning from Demonstrations that leverages small sets of demonstration data to significantly accelerate the learning process in RL. (Levine et al. 2016) presented a learning-based approach to hand-eye coordination for robotic grasping from

monocular images. The authors trained a large convolutional neural network to predict the probability of successful grasps based on the task-space motion of the gripper, using only monocular camera images. However, this approach assumes that the goal state and position are predefined. To overcome this limitation, our method learns the goal state from the collected demonstrations, which differs from approaches that rely on predefined goals and positions.

Methodology

Figure 1 shows three main modules designed for efficient trajectory generation and state transformation. The first module, Keyframe Identification, constructs a training data repository by employing dynamic programming to extract keyframes from the input trajectory, capturing essential temporal instances. The second module, Goal-Conditioned Keyframes-Guided Trajectory Generation, predicts the final state based on initial RGB images and point cloud data using an encoder-decoder structure to optimize computational resources. The goal generation network within this module utilizes Kullback-Leibler Divergence (KLD) to assess predictive accuracy. The third module, a goal-conditioned policy learning framework, is bifurcated into an imitation learning component, where a convolutional neural network predicts subgoals, and a reinforcement learning component, where a Soft Actor-Critic (SAC) algorithm-based policy generates actions for state transformation. Trained on classified keyframes, this integrated approach ensures enhanced learning efficiency and trajectory generation.

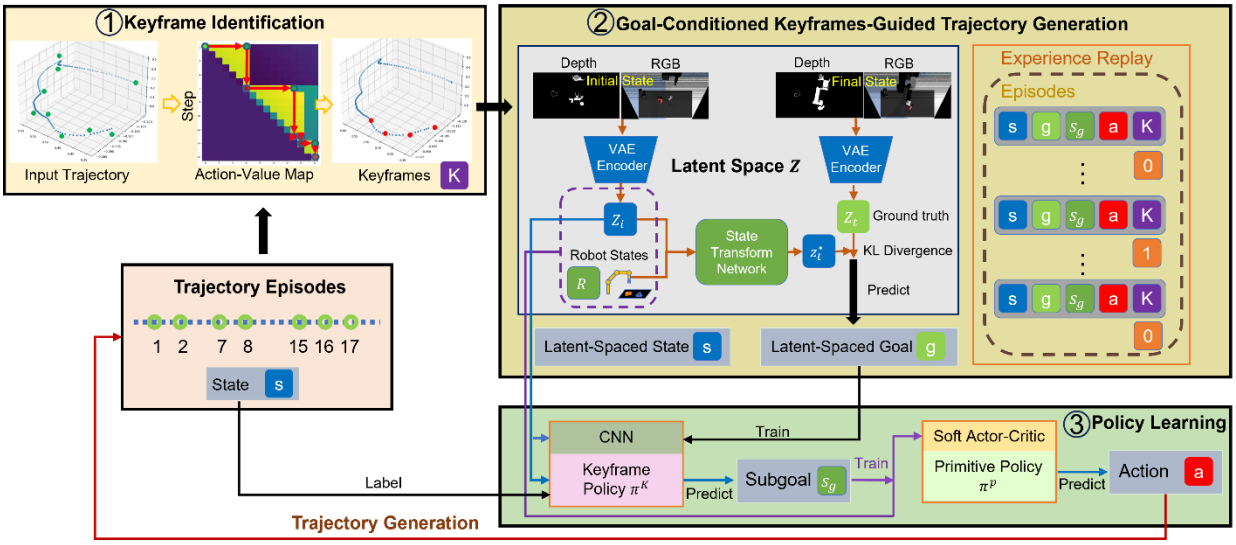

Figure 1. Structure of our method.

The workflow of the algorithm, detailed in Algorithm 1, proceeds as follows: the imitation learning approach trains both the keyframe generation policy and the keyframe evaluation system using a collection of demonstrations. Following this, the primitive policy is iteratively updated, using the generated keyframes as directional objectives for executing basic motion sequences. The agent's generated trajectories are evaluated by the keyframe evaluation module. Those evaluated as proficiently executed are added to both the experience replay buffer and the imitation episode buffer. Furthermore, the latent space module generates latent states relevant to both the imitation and reinforcement learning modules. Subsequent subsections will provide a detailed elaboration on each constituent element, exploring the intricacies of keyframe generation, evaluation, latent space computation, and algorithmic refinement.

Algorithm 1 Easy Teaching Algorithm

1: $\boldsymbol{N}$:= max episodes, $\boldsymbol{M}$:= max steps for each episode
2: Load the pretrained encoder and decoder for the latent space network
3: Initialize keyframe generation network $\boldsymbol{\pi_k}$
4: Initialize keyframe classification network $\boldsymbol{C_\psi}$
5: Initialize primitive motion network $\boldsymbol{\pi_p}$
6: Initialize Experience Replay Buffer $\boldsymbol{B_E}$
7: Initialize Imitation Episode Buffer $\boldsymbol{B_I}$
8: **for** $\boldsymbol{i}$ from 1 to $\boldsymbol{N}$ **do**
9: Generate the goal state $\boldsymbol{g}$ for current state $\boldsymbol{s}$

10: **for** j from 1 to M **do**
11: Generate subgoal s_g with s and g utilize π_k
12: Generate action a with s and s_g utilize π_p
13: Collect the state s' after execution and reward r
14: Update Primitive policy π_p
15: Put the current data to the experience reply buffer B_E
16: Add the collected data to current episode trajectory T_i
17: **end for**
18: Evaluate trajectory T_i and identify the keyframes T_i^k
19: Generate Imitation learning items from T_i and T_i^k, add them to Imitation Episode Buffer BI
20: Update the parameters in π_k and C_ψ
21: **end for**

Keyframe Guided Trajectory Generation

Figure 2 shows our policy learning architecture composed of key components that collaborate synergistically: keyframe policy π_k, primitive policy π_p. Latent-Spaced Goal g and Latent-Spaced State s, coming from trajectory generation module, are fed into fed into a Convolutional Neural Network (CNN) serves as Keyframe Policy π_k, which interprets the information and predicts a subgoal s_g. Latent Vector z, coming from depth and RGB images encoded by the VAE encoder, is fed into the CNN as well. The subgoal, together with the current Robot states R, are used to train a Soft Actor-Critic Primitive Policy π_p, responsible for robot action prediction. The output of this architecture is a robot Action a, which is the actual command that would be executed by the robot.

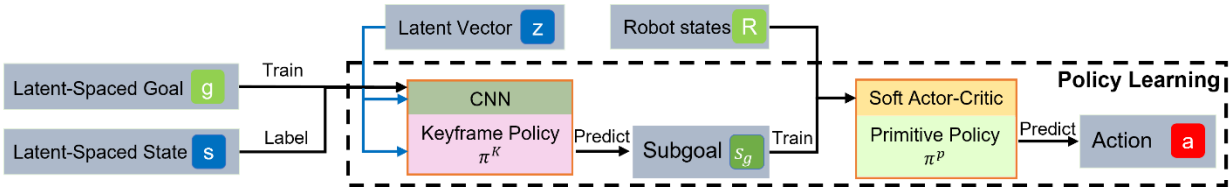

Figure 2. Policy learning architecture.

A significant hurdle in our field is balancing the need for robotic learning against the scarcity of experts in robotics algorithms. Our model mitigates this by adopting a novel three-network training architecture. This framework consists of a keyframe policy, primitive policy, and keyframe classifier, all integrated within a GAN setup. This unique configuration facilitates the transfer of human skills to robots,

enabling them to generate dynamic subgoals and adapt to the construction environment's complexity and unpredictability. Furthermore, our keyframe identification model sharpens the focus on significant task moments, optimizing learning efficiency. This approach not only boosts construction robot efficiency and autonomy but also adeptly navigates the nuances of safety, environmental variability, and the seamless integration into existing construction processes.

Keyframe Policy: Keyframe Generation

The keyframe order in the trajectory should be considered when teaching the robot. In reinforcement learning tasks, there are many cases with an unknown number of subgoal generations. To address this problem, we formulate the trajectory distribution as an MDP function which naturally encodes the keyframe order into the trajectory.

$$p(s_0, g) = \prod_{t\sim\tau} p(s_t, g, a_t) p(a_t | s_t, g) \quad (1)$$

Then the expected discounted return of the trajectory is

$$R = E_{t\sim\tau}[\sum_a \gamma^t r(s_t, g, a_t)] \quad (2)$$

Where $r(s_t, g, a_t) = 1[s_{t+1} == g]$. A randomly initiated policy is not feasible for providing a successful trajectory, resulting in consistent negative rewards and no effective training. Expert demonstrations are introduced to provide a guideline for the robot to train in the right direction, and the trajectory distribution is calculated using the keyframe extraction method.

Directly generating all keyframes for the trajectory conditions on the initial state s_0 and goal g makes it hard to guarantee the success of the task. To mitigate compounding errors, we propose generating the keyframes following an MDP process that conditions the generation on the current state s_t and goal g. Then the policy can be optimized by maximizing the expected discounted reward.

$$J(\theta) = E_{g\sim\rho_g, \tau\sim\tau(.|g)}[\sum_t \gamma^t r\big(s_t, s_{g_t}, a_t, g\big)] \quad (3)$$

The keyframe is a distribution condition on the current state s_t and goal g, we add a classification $C_\psi(s_t, s_{g_t}, a_t, g)$ network to distinguish the keyframe and ordinary states. The classification network takes the current state, generated subgoal, action, and goal state as input and provides a unique label to indicate whether the sampled frame is a keyframe or not. To utilize the keyframe demonstration information in the reinforcement learning loop, we add the expected keyframe score as the regularization part of the expected discounted return function $J(\theta)$.

$$J(\theta) = E_{g\sim\rho_g,\tau\sim\tau(.|g)}[\sum_t \gamma^t r(s_t, s_{g_t}, a_t, g)] + E_{s_{g_t}\sim\pi^K, a\sim\pi^P}[C_\psi(s_t, s_{g_t}, a_t, g)] \quad (4)$$

where $E_{g\sim\rho_g,\tau\sim\tau(.|g)}[\sum_t \gamma^t r(s_t, s_{g_t}, a_t, g)]$ is the expectation of the demonstration score, $g\sim\rho_g$ is the goal distribution, and $\tau\sim\tau(.|g)$ is the demonstrated trajectory, $E_{s_{g_t}\sim\pi^K, a\sim\pi^P}[C_\psi(s_t, s_{g_t}, a_t, g)]$ is the expected keyframe score parameterized by ψ, π^K is the keyframe generation policy, π^P is the primitive policy.

Different from generative adversarial methods (Ding et al. 2019; Ho and Ermon 2016), our classification method assigns scores to sampled frames instead of distinguishing them from those generated by experts or policies. In our classification network, we employ a regression-style layer instead of an activation layer, assigning positive and negative labels with 1 and -1, respectively. By assigning positive and negative labels, we penalize frames that are not keyframes and encourage agents to generate keyframes. The regression layer, being the last layer, takes the keyframe distribution into account. The keyframes extracted by the keyframe extraction method are within a keyframe distribution, which cannot guarantee optimal keyframes but ensures they are not far from being optimal. Therefore, we use the extracted keyframe as the label baseline. There could be better keyframes with scores greater than 1, and frames significantly worse with scores less than -1. The loss of the classifier is the regular cross-entropy loss, and the loss for the expert keyframe imitation is:

$$L_{C_\psi} = |E_{s_g\sim K}[C_\psi(s, s_g, a, g)] - E_{s_g\sim\pi^K}[C_\psi(s, s_g, a, g)]| \quad (5)$$

Where K is the demonstrated keyframe set and π^K is the keyframe generation policy. By taking this loss, we connect the demonstrations and the keyframe generation policy.

Primitive Policy: Goal-Conditioned Soft Actor-Critic

Since the primitive policy is working on the same agent, it shares the same state space S, action space A, and environment dynamics P. Then the formulated finite-horizon, goal-conditioned Markov decision process can be defined by tuple $M_p = (S, A, P, g_p, r_p, \gamma_p)$, where S, A, P are the same with M and the reward function is $r_p(s, a, g)$ the discount factor is γ_p. The primitive policy is formulated into a regular goal-conditioned soft actor-critic reinforcement learning framework. And the primitive policy is fitted by maximizing the expected discounted return:

$$J_\xi(\pi^p) = E_{g_p\, \rho_g, \tau_p\, d^{\pi_p(.|g)}}[\sum_t \gamma_p^t r_p(s_t, a_t, g)] \tag{6}$$

with the trajectory distribution

$$d^{\pi^p}(\tau|g) = \rho_0(s_0) \prod_t \pi^p(a_t|s_t, g) p(s_{t+1}|s_t, a_t) \tag{7}$$

where $J_\xi(\pi^p)$ is the expected discounted reward of the primitive policy and ξ is to distinguish with the keyframe generation reward. And $\pi(.|s, g)$ generates continuous robot action conditioned on state s and goal g. The primitive policy is updated standalone and follows the standard off-policy actor-critic paradigm. There are two phases of training the primitive policy: pretrain and train along with the whole model. During the pretrain phase, we take advantage of the hindsight experience replay technique to accelerate the training process. The action works on the transition of tuples (s_t, a_t, s_{t+1}, g), and the critic evaluates its action-state value. The update function is in the following equations concerning Q-function parameters ϕ_{k+1}

$$Q_{\phi_{k+1}} = argmin \frac{1}{2} E_{(s_t, a_t, s_{t+1}, g)D}\left[r_t - Q_\phi(s_t, a_t, g)\right]^2 \tag{8}$$

With the target value

$$r_t = r(s_t, a_t, g) + \gamma E_{a_{t+1}\pi(.|\, s_t, g)} Q_{\phi_k}(s_{t+1}, a_{t+1}, g) \tag{9}$$

The policy update by maximizing the discounted reward respect to advantage function $A^{\pi}(s,a,g) = Q^{\pi}(s,a,g) - V^{\pi}(s,g)$, where $V^{\pi}(s,g)$ is the value function for current policy.

$$\pi^{*} = arg\max_{\pi} E_{(s,g)\sim D, a\sim\pi(.|s,g)}[\sum_{t} r(s_t, g_t, a_t) + \alpha \log \pi(\cdot\,|s_t)] \quad (10)$$

Keyframe Extraction Method

The collected trajectories are sampled directly from the robot state broadcasting. As in our case, the robot states broadcasting at 200Hz. Therefore, for each trajectory, the collected data is much more redundant than necessary. Furthermore, the execution of robot manipulation tasks generates trajectories, whether they are derived from human teleoperation or reinforcement learning explorations. However, these trajectories, irrespective of their source, do not guarantee optimality. This section aims to tackle the issue of noise inherent in trajectories generated by human-operated and the reinforcement learning agent-operated robotic arms. To mitigate irrelevant movements and emphasize essential information, we introduce the concept of keyframes. Keyframes are defined as minimal sampled poses of the robot end-effector within a trajectory that enable task completion. To eliminate the redundant road points in the trajectory of the succeeded task, and avoid computation explosion problem, we extract the keyframe with two steps: trajectory down sampling and essential keyframe identification from the down sampled trajectory. This method ensures efficient use of data while maintaining trajectory accuracy.

Frame Sampling

The collected states in the demonstration episode are $S = \{I, D, p\}$, I is the RGB image, D is the depth image, and p is the robot and the end-effector position. $p = \{j_1, j_2, j_3, j_4, j_5, j_6, e_p\}$, where $j_n, n =$ 1,2,3,4,5,6 is each joint angle of the robot, e_p is the end-effector states. For each trajectory, the collected robot states history can be expressed as

$$T = \{p_0, p_1, \ldots\ldots, p_n\} \quad (11)$$

where the p_n, n = 1, 2, 3, . . . is the sampled robot states (road points) from task execution trajectory. Therefore, we have a collection of line segments.

$$L = \{l_0, l_1, \dots\dots, l_{n-1}\} \tag{12}$$

We claim that, in straight-line segments, all road points except the endpoints are redundant. We extend this assumption to the angle between consecutive trajectory segments, proposing that road points within a certain angle threshold are considered redundant. Furthermore, we assess the angle between every two consecutive line segments to identify turns. We assume that the turns are more important than other road points in the trajectory. Then

$$a = \frac{l_i l_{i+1}}{|l_i||l_{i+1}|}, \theta = arcco\,s(a)\,, i + 1 < n \tag{13}$$

Where l_i, l_{I+1} are the line segments on the trajectory, a is the cosine value of two vectors, and θ is the angle between l_i and l_{i+1}. Then we apply the peak finding method (Kadane 2023) to find the local maximum α as the sampled actions from the expert trajectory. We define the sampled trajectory as

$$T_s = \{p, p_2, p_3, \dots\dots, p_m\}, m \leq n \tag{14}$$

In scenarios where frames exhibit a peak angle and cannot be keyframed, especially when incorporating human demonstrations, the influences of operational habits or other noise are common. These frames frequently result in longer and less efficient routes for the robot, sometimes introducing setbacks. To address this issue, our approach not only extracts these peak angle frames but also uniformly samples additional frames from each segment of the trajectory. These extra frames are added to the keyframe candidate set, enabling a more thorough and effective keyframe selection that accommodates and reduces the impact of such irregularities.

We introduced T_s' to denote the enhanced set of key states, which includes not only the peak angle frames identified through the peak finding method, but also additional frames uniformly sampled from each

segment of the trajectory. This enrichment of T_s into T_s' aims to mitigate the impact of operational habits or noise, further optimizing the trajectory for learning.

$$T_s' = \{p_1', p_2', p_3', \dots\dots, p_m'\}, m \leq n \tag{15}$$

Keyframe Identification

Our method focuses on identifying keyframes essential for forming the optimal trajectory in the robotic trajectory refinement process. Although many points are initially removed, the remaining ones are pivotal for the trajectory's progression. To optimize the trajectory, we define keyframes as frames that constitute the most efficient path. We employ a reinforcement learning-based approach to identify these keyframes from the collected trajectory, aiming to further reduce the number of sampled frames. This strategy substantially increases the likelihood of a successful robot operation. The keyframe identification process experiments with subsets of the trajectory T_s to identify the most effective subset as the keyframe. This concept is represented in Figure 3, where Fig. 3a shows the trajectory's keyframes and Fig. 3b shows a blue line representing the trajectory. Green dots represent human control input samples from a demonstration; these are fewer than the trajectory frames and typically deviate from the path. Red dots represent extracted keyframes for the robot manipulator, underscoring the crucial points for efficient trajectory control. This module is dedicated to the construction of a training data repository, pivotal for the subsequent learning processes. It employs dynamic programming to meticulously extract keyframes from the input trajectory, thereby encapsulating significant temporal instances that are instrumental in characterizing the trajectory.

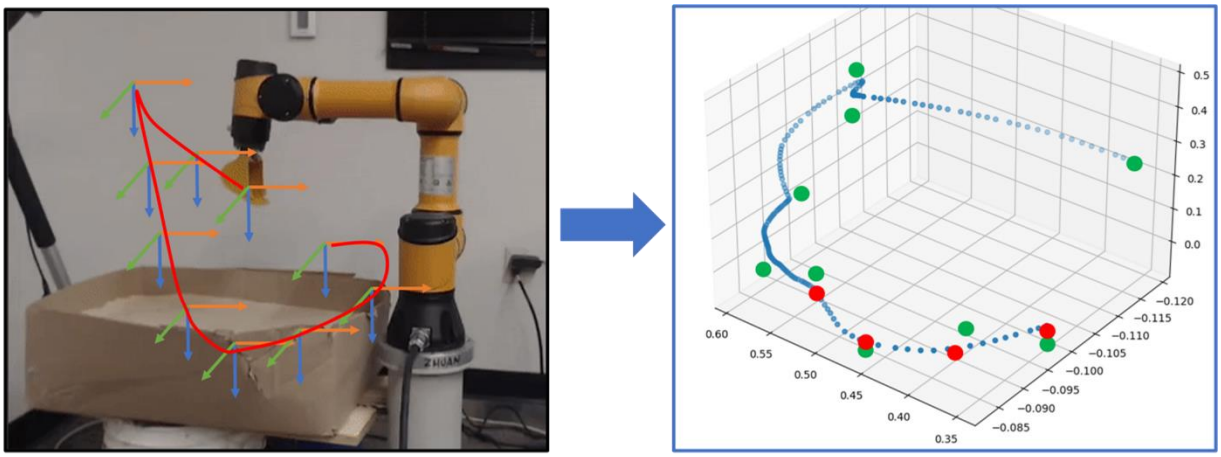

Figure 3. Extracted keyframes. The Fig. a shows an excavation task example, while the Fig. b displays the collected trajectory in 3D coordinates. Blue dots represent actual robot states, green dots represent human teleoperated control inputs, and red dots signify extracted keyframes. Green dots appear sporadically due to the nature of human teleoperation, where new commands are issued before the previous one is completed.

To identify the optimal keyframe from the candidate set, we omit frames from the sampled frames T_s', and execute the remaining frames to observe the task execution outcome. Then we rank all possible combinations to identify the best subset as the keyframe set T_k.

To evaluate the different combinations, we employ the Markov Decision Processes (MDP) (Fang et al. 2018) model to formulate the problem into a reinforcement learning context. The goal of this MDP model is to collect a minimal subset of T_s' capable of completing the task. To reduce the sampled frames, we assign a small negative reward for each frame where the task is incomplete, and a positive reward is given upon task completion. By using a greedy policy to select the subset with the largest reward as the keyframe collection, we can eliminate as many low-weight samples as possible. This is because the negative reward decreases the total reward for unnecessary samples. We model the action set as

$$\pi(s) = a \tag{16}$$

where s is the current state, a is the action taken at the state s, and s' is the state after s takes action a.

According to the Bellman equation, the value function for each state $V^{\pi}(s)$ is shown in Equation 17. The return G_t is defined in Equation 18. We assign the uniform distribution for the initial policy.

$$V^{\pi}(s) = E_{\pi}[G_t|S_t = s] \tag{17}$$

$$G_t = \sum_{\{i=0\}}^{\infty} \gamma^i R_{t+i+1} \tag{18}$$

$$V^{\pi}(s) = E_{\pi}[R_{t+1} + \gamma G_{t+1}|S_t = s] \tag{19}$$

$$V^{\pi}(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)[r + \gamma V^{\pi}(s')] \tag{20}$$

To avoid the waste of unnecessary computation, for each frame, we assume the previous frame is succeeded and this results in the backward drop frames, which is the depth-first search problem. The optimal keyframe set is selected according to the maximum collected rewards.

$$a = arg\ \underset{a \in A}{argmax} \sum_a \pi(s) \tag{21}$$

Latent space exploration and goal generation

Directly using images as inputs presents challenges. First, integrating image inputs can introduce redundancy, potentially leading to slow convergence. We introduce an approach based on latent space exploration to reduce dimensionality, providing more precise information, and reducing the unnecessary burden on subsequent models, thereby enhancing the efficiency and effectiveness of training.

Second, insufficient demonstration data can impede the system's ability to learn effective policies or induce overfitting. To tackle this, we employ a strategy enabling continuous and effective learning to ensure the sustained operation of the system. We propose a goal state generation model that learns from demonstrations. In this model, the initial state of each episode is represented by its first frame, and the goal state by the last frame, to train the goal generation network in the latent space. Additionally, the goal generation network generates the goal latent state based on the initial latent state. The model's architecture, a modified version of the beta-VAE model (Higgins et al. 2016), is illustrated in Figure 4. The latent space is crucial for reducing dimensionality and enhancing the model's accuracy and efficiency. However,

conventional methods such as principal component analysis (PCA), autoencoders, and K-means may overlook critical details essential to other models. To address this limitation, we propose a modified beta-VAE model as depicted in Figure 4. The latent space module is ingeniously designed to predict the final state based on the initial state, which encompasses RGB images and point cloud data. It leverages an encoder-decoder architecture to efficiently transform the RGB image and point cloud into a compressed vector representation, thereby optimizing computational resources. The compressed vector is subsequently input into a goal generation network, which endeavors to predict the final state. The fidelity of the prediction is quantitatively assessed using the Kullback-Leibler Divergence (KLD), facilitating the evaluation of the network's predictive accuracy. The outputs of this module are the latent-spaced state and the latent-spaced goal, which serve as critical inputs for the succeeding module.

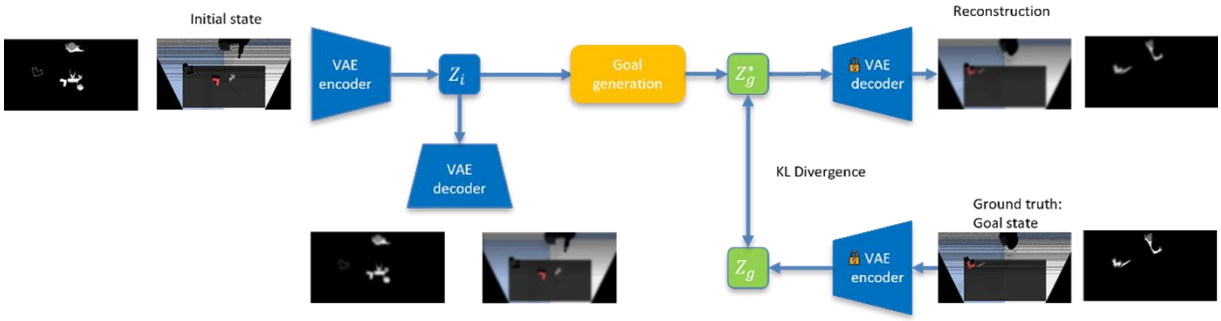

Figure 4. Latent space goal generation network

To validate the effectiveness, we conduct an early experiment to show the results of the generation tryout in Figure 5. Our network exhibits convergence at approximately 600 steps, with the subsequent error remaining stable thereafter. Since this error pertains to the KLD error in the latent space, there is no specific unit associated with it. The error evaluates the similarity between two distributions, with lower error indicating greater similarity between the distributions. The parameters for our training process included a learning rate of 0.001, a batch size of 64, and we trained our model for 1000 steps. The latent space dimension was set to 50, providing a balance between model complexity and computational efficiency. We used a β-VAE with β set to 0.5 to encourage disentangled latent representations while still prioritizing

reconstruction accuracy. Our optimizer of choice was Adam, with a dropout rate applied of 0.5 for regularization to mitigate overfitting.

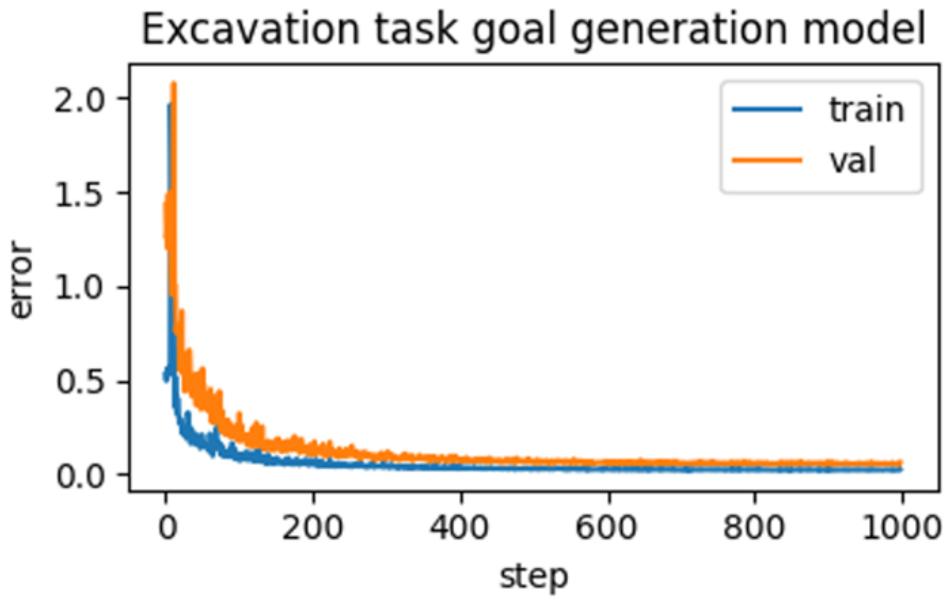

Figure 5. The loss of goal generation of excavation tasks.

Experiments and Evaluation

Implementation details

Robot Teleoperation Framework Setup

Given the high cost and operational dangers of a real excavator, we simulate it using a robot platform. This approach allowed us to explore advanced control systems and the flexibility necessary for precise and controlled research tasks, which are critical for advancing construction robotics. Unlike standard excavators, our robot offers enhanced maneuverability and an intuitive control system, enabling detailed investigation into automated tasks that are challenging with traditional machinery. This choice not only facilitates research into automation technologies adaptable to various construction equipment, potentially lowering costs and improving site efficiency, but also addresses safety concerns by reducing the risks associated with direct human interaction with heavy machinery. A bucket is attached to the end

effector to simulate the excavator's shovel. Sand is placed in a 6-inch-thick box to simulate soft soil. The robot is controlled using a VR controller (HTC VIVE). The setup for the teleoperation system is shown in Figure 6.

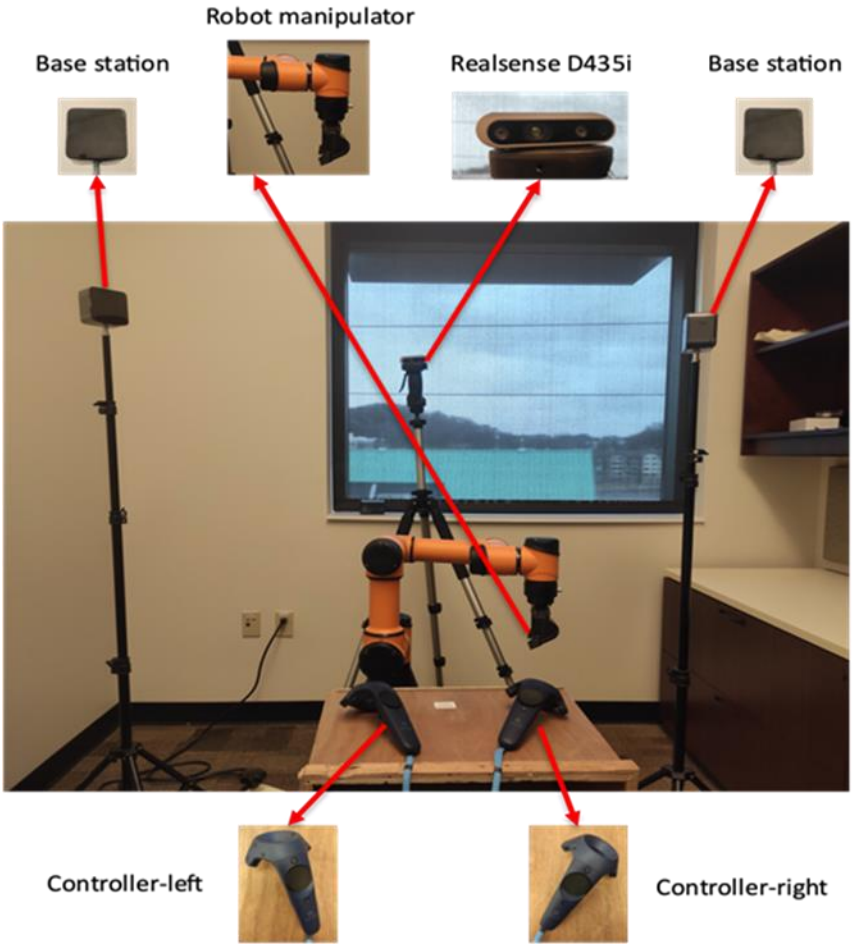

Figure 6. Excavation task system setup.

This architecture consists of four main components. Initially, the process involves extracting and relaying the HTC VIVE controller's position data to the Robot Operating System (ROS) using Steam VR Beta and the HTC VIVE SDK. The controller's coordinate origin is shifted from the top to the bottom for intuitive operation. Next, the controller's position and movement data are processed using an inverse kinematics algorithm to determine each robotic arm joint's target position. This approach, which relies on

changes in position and rotation rather than absolute positioning, provides the operator with greater flexibility regarding their standing location. Meanwhile, joint data can be accessed through ROS. In the third stage, ROS Rviz displays the robotic arm's trajectory, following the received joint instructions. This setup not only facilitates control execution but also aids users in evaluating their control strategy. Lastly, a UDP protocol links the ROS nodes with the actual robotic arm, enabling operators to control a physical robot by publishing modified data. The modified data that is published via the UDP protocol to control the physical robot refers to the calculated joint angles and positions necessary for the robotic arm to replicate the movements dictated by the HTC VIVE controller. This data is essentially what bridges the operator's intentions with the robot's physical actions, enabling intuitive control over the robotic arm's movements.

Experience replay buffer and Imitation Episode Buffer

We utilize the experience replay buffer technique incorporating specific adaptations tailored to our methodology. Each item in the buffer is carefully designed to include the following components: an initial state randomly sampled from the trajectory, a subsequent keyframe as the subgoal, a final frame as the goal state, and an associated keyframe label as the reward for the primitive policy. During the training process, these components serve distinct roles: the subgoal aids in refining the keyframe generation model, while the other elements are crucial in sharpening the primitive policy. This strategic approach guarantees a cohesive learning process, effectively integrating keyframe generation with primitive policy training.

Similarly, when constructing the Imitation Episode Buffer, we adopt a process similar to the Experience Replay Buffer. Here, we select states from the trajectory, including the nearest keyframe adjacent to the selected state and the final goal, integrating them into a single keyframe generation item. For the Experience Replay buffer, we prioritize shuffling the buffer’s order to mitigate inter-item dependencies and foster robust learning.

Gravity axis alignment

Figure 7 highlights the importance of z-axis alignment, crucial for maintaining consistent directionality along the z-axis. This alignment is vital for ensuring that the point cloud from the RealSense

camera seamlessly matches the robot manipulator's z-axis orientation. This alignment is achieved through synchronization with the direction of gravity. The effectiveness of this alignment is demonstrated in Figure 7.

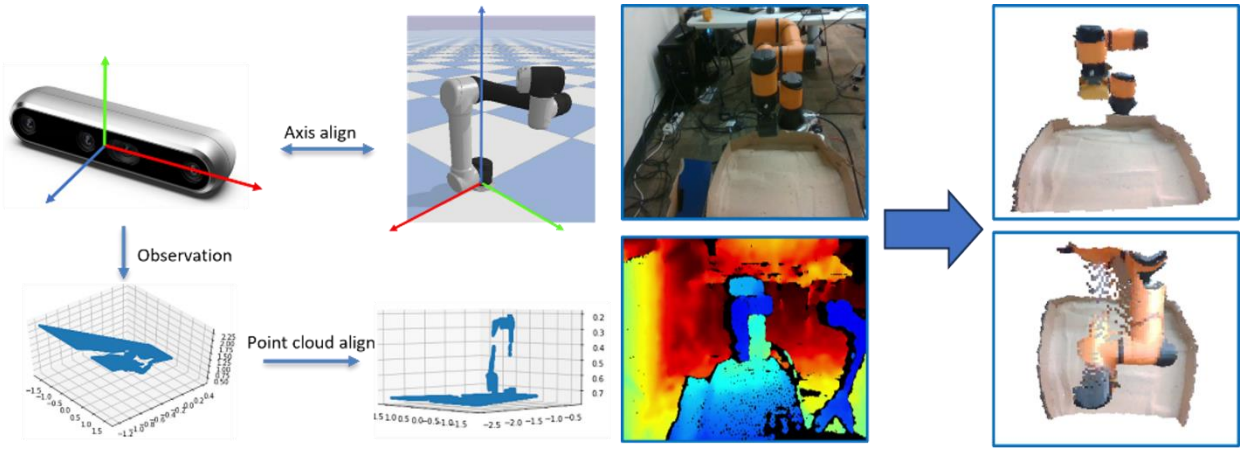

Figure 7. Gravity axis alignment.

Recognizing the critical role of accurate alignment in ensuring the quality of robot learning data, our choice of this method was driven by its potential to precisely capture spatial relationships essential for robotic perception and action. This method allows for a detailed representation of the environment, facilitating more accurate robot interactions with complex and dynamic construction settings. While our initial methodology did not explicitly quantify numerical errors associated with this calibration, it was predicated on the adaptability of deep learning models to effectively interpret and utilize imperfect data. Deep neural networks, by design, can identify and mitigate the impact of data irregularities, including those introduced by calibration discrepancies. This inherent robustness to noise and bias makes the method particularly suited for environments where precision and adaptability are paramount. Incorporating this calibration approach, coupled with the sophisticated error-correction capabilities of deep learning, ensures that our robot learning framework remains resilient against minor misalignments and inaccuracies. This synthesis of advanced calibration techniques and neural network processing addresses both the immediate

needs of robotic learning in construction and the long-term goal of creating autonomous systems capable of operating within highly variable and unstructured environments.

We use Inertial Measurement Units (IMUs) for z-axis alignment, providing a more precise and efficient approach for parameter estimation and spatial correlation between the robot and camera. Leveraging IMUs simplifies and increases the accuracy of the alignment process, enhancing its efficiency and reliability.

Keyframe-guided Trajectory Generation

This section presents the results of experiments. We change the input data from engineered features to raw, gravity-aligned point clouds. Although organized in image order, pixels in the point cloud exceeding a certain depth threshold are set to 0. The experimental procedure unfolded in several systematic phases to ensure rigorous testing and accurate results. It commenced with the initialization phase, where the robot was set to a standardized starting position and primed to perform the excavation task. In the task execution phase, the robot employed each method under scrutiny to complete a pre-defined excavation task within the simulation environment, which involved the translocation of materials. During the performance measurement phase, we meticulously monitored and logged the robot's precision, efficiency, and adherence to safety protocols against our task success criteria. This multi-step process was replicated over 10,000 trials for each method, providing a robust dataset and allowing for the assessment of performance consistency across trials. A trial was deemed successful if the robot accurately accomplished the excavation task according to the predetermined criteria, operating autonomously without human intervention. Finally, in the data analysis phase, we computed the success rate for each method by calculating the ratio of successful trials to the total number of trials, with these results informing the comparative effectiveness of each method tested. The results, shown in Figure 8, demonstrate that our method achieved the highest success rate among the compared methods. The left figure shows the method not utilizing latent space, with the goal state manually provided. The right figure illustrates our method. Despite yielding a slightly lower success rate compared to the method without the latent space, we note that our method operates without

manually specifying the final goal state. This supports our claim of reducing reliance on robotic or algorithmic expert involvement, thus highlighting the autonomy and self-sufficiency of our method.

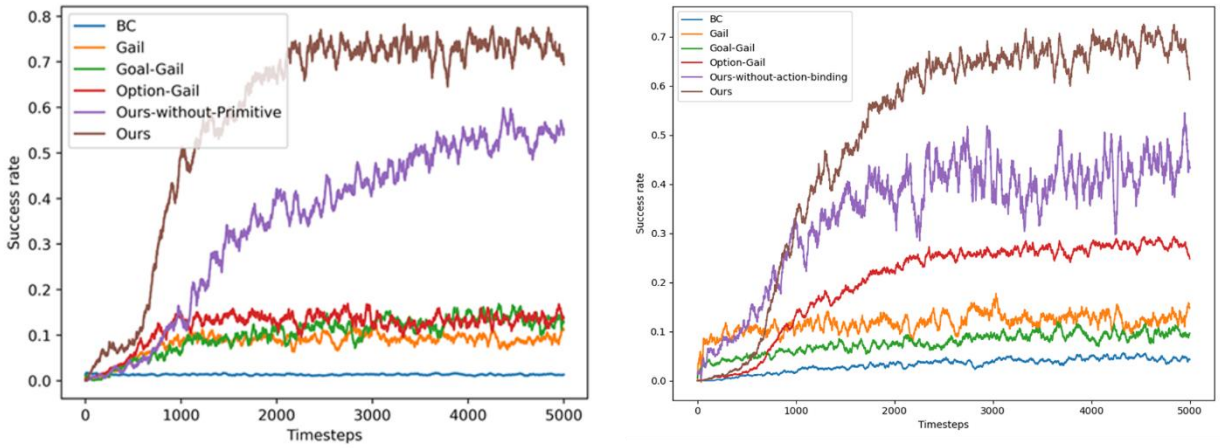

Figure 8. Task success rate. The figure 8.a illustrates our algorithm without latent space representation, where the goal and subgoal states are manually designed and not represented by images. In contrast, the figure 8.b depicts the success rate throughout the training process using latent space representation, eliminating the need for manual intervention. In the 8.a, the blue line stands for the behavior cloning method minus -0.01 to distinguish it from the Gail method success rate.

To validate our proposed model, we conduct an experiment on a simulation platform to estimate the robot end-effector's position. We compare our method with several benchmarks, including the vanilla network, the spatial SoftMax method proposed by (Levine et al. 2015), and the beta-VAE method. In the experiment, we use aligned point cloud data with color as the network input, which then generated the output for the robot end-effector's position. The results, presented in Figure 9, show the loss in regular Cartesian coordinates at the top, while the bottom figure uses a logarithmic axis to depict the same loss for enhanced clarity. Notably, while the training loss for each method remains relatively consistent, the validation loss shows significant fluctuations. Using a logarithmic axis in the bottom figure further highlights the distinctions between the losses of each method. Our method establishes a stronger correlation

between the encoded images and robot actions than the other methods evaluated. This demonstrates the efficacy of our method in accurately estimating the robot end-effector's position.

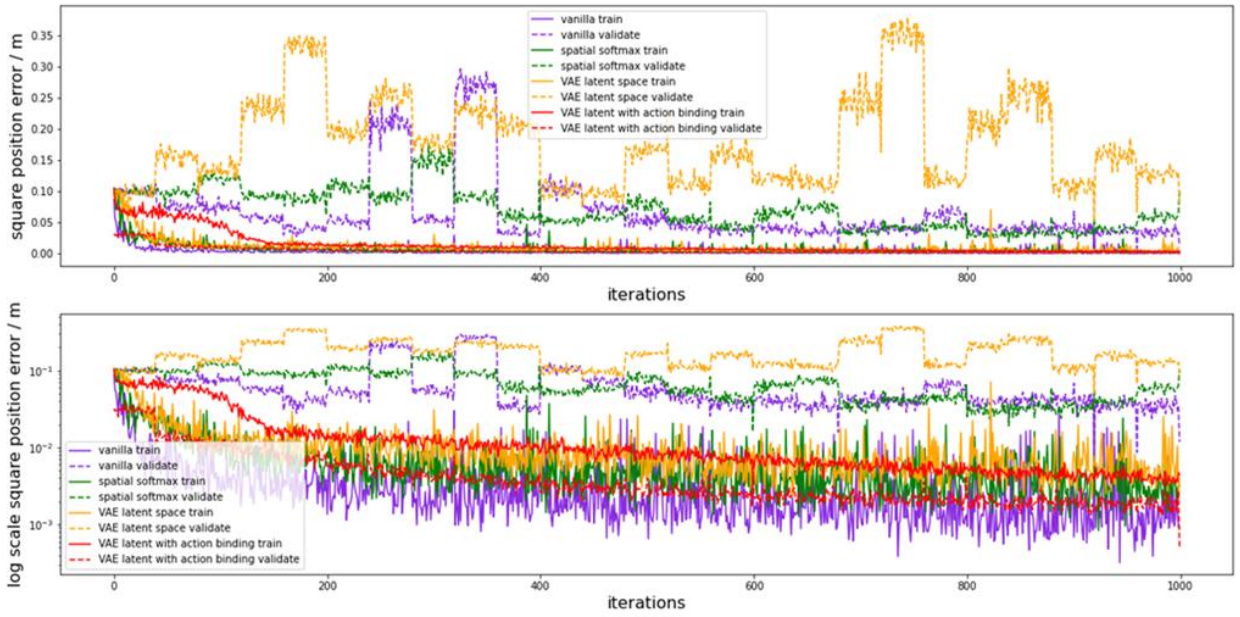

Figure 9. Robot end-effector position estimation.

Ablation study

We conduct an experiment to study the influence of latent dimensions, testing sizes of 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100. In addition, we perform experiments in both simulated and real excavation scenarios, with and without our proposed binding model. The results, presented in Figure 10, show that the experiment with our proposed binding model outperforms the experiment using the pure auto-encoder method. We choose 30 as dimension to compare across dimensions and considering both outliers and the success rate. While dimension 80 is also viable, we prefer a smaller dimension for our model. This is because a larger dimension introduces more uncertainty, leading to increased estimation variance. (Chen and Storey 2015) addresses the problem of extracting low-dimensional structures from high-dimensional data and discusses how under certain conditions, it is possible to consistently recover the structure using information up to the second moments of these variables. It implies that when attempting to model or

estimate using higher dimensions, the complexity and uncertainty increase, which can affect the accuracy and variance of the estimations. This aligns closely with the concept that larger dimensions introduce more uncertainty and increase estimation variance. In Figure 10, the success rates with our proposed binding model are nearly 20% higher than those without the binding model. This result demonstrates the significant role our proposed binding model plays in the training process, improving the success rate for the excavation task.

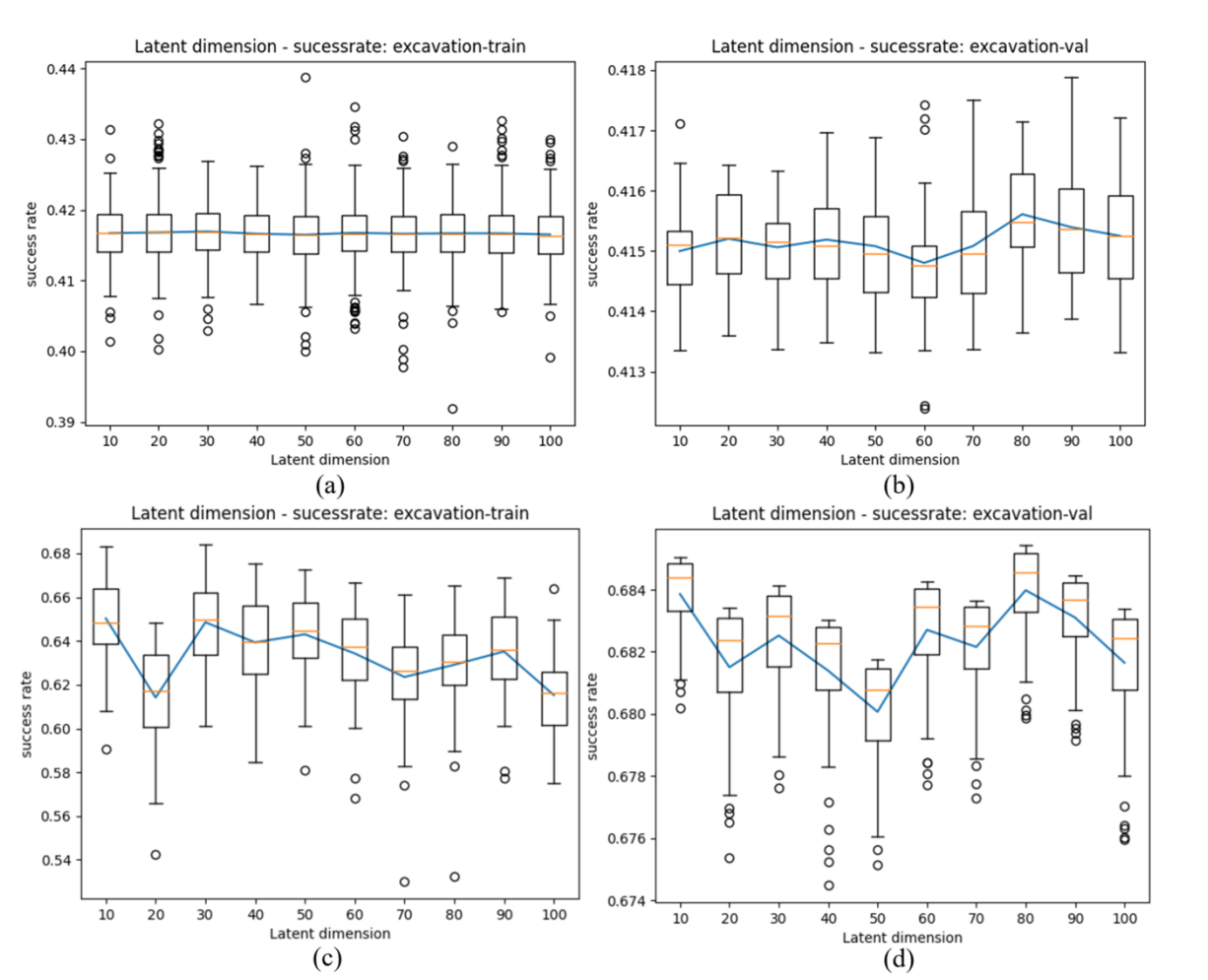

Figure 10. Latent space dimension study for excavation task. (a/b) Train/Validation of excavation task without binding model; (c/d) Train/Validation of excavation task with binding model.

Discussion

618 The applicability and scalability of this research are shown in three aspects. First, most of the time
619 the construction industry is reluctant to change, posing significant challenges to collect data from expert
620 demonstrations in construction sites. Using the proposed teleoperation method as an intermediate solution
621 can increase the feasibility, desirability, and viability of the construction industry to use semi-automated
622 robots. This method can also help to collect the necessary data to train the robots towards full automation.
623 With the operator involved in the teleoperation process, the knowledge, abilities, and expertise of the
624 operator can be used to train construction robots. The use of the operator skills helps in retaining the security
625 and efficiency of construction sites. Expert demonstration collected from teleoperation is significantly
626 different from simulation and VR demonstration. Unlike simulated data, expert demonstration provides
627 more realistic and practical data. This valuable data enables robots to be trained more effectively. In the
628 future, such training will prepare construction robots to handle complex tasks in unstructured sites. Second,
629 as demonstrated by the keyframe extraction results, the proposed method can eliminate over 80% of the
630 redundant frames in the expert demonstration. This achievement can streamline the data required for robot
631 training, reducing computational loads and improving efficiency. The well-known "garbage in, garbage
632 out" philosophy highlighted that the quality of data greatly influences the quality of the results. To ensure
633 that robots can imitate human expert control in complex construction processes, it is essential to remove
634 noise and redundant frames from the expert demonstration data. This is particularly important for
635 construction robots. In construction, human operators are often disturbed, and the data they generate tends
636 to have more substantial noise compared to other operations. The proposed keyframe method employed the
637 RL-based method to find the optimal subset as the keyframe set, which was demonstrated in various
638 sampling tasks. In addition, the learning results showed that because of the keyframe extraction, robot
639 learning becomes very efficient and effective, and is superior to state-of-the-art methods. After acquiring
640 the keyframe, a model-free robot training method based on keyframe extraction, and a hierarchical imitation
641 learning method were proposed. The stochastic primitive policy is pre-trained with soft actor-critic and with
642 hindsight experience replay (HER) method. To demonstrate the applicability and scalability of our method,
643 two scenarios were evaluated, where construction robots can be widely used: excavation in both simulation

and real-world settings using different robotic arms. These demonstrated the generalizability of the proposed method concerning robotic trajectory tasks and robotic manipulators. Such generalizability is critical for the application of robots for full automation in the construction industry, a fact that is reinforced by the superior results obtained compared to the state-of-the-art. Third, to enhance the extensibility to a wider range of tasks and reduce the computational loads of RL, an integrated approach that combines vision-based trajectory generation with latent space exploration was proposed. Specifically, the raw states were substituted with latent states, and the primitive policy was pre-trained using a latent space variable and latent space states were compiled in the trajectory buffer. The obtained result shows that our proposed binding model outperforms the pure auto-encoder method. For the excavation task, the binding model plays an important role in the training process, boosting the success rate from 40% to over 60%. The result demonstrates that latent space exploration facilitates the training process of RL policy and improves robot learning performance.

This research has several limitations that deserve future research. First, a significant limitation of imitation learning is that the robot's proficiency can only match the quality of the expert demonstration. Specifically in the context of construction sites, many tasks hinge on human operators' subjective assessments and experiences, and their execution is not necessarily optimal. Therefore, there can potentially be a substantial scope for enhancing the robot's manipulative abilities in complex construction environments. In addition, imitation learning trained robots do not always generalize well to scenarios that were not included in the training data. The construction site dynamics further pose significant challenges for such robots trained by imitation learning. In the future, novel machine learning methods should be integrated with sensing and engineering knowledge. This integration will equip robots with robust performance in various scenarios. Additionally, it will help optimize robot's trajectory and manipulation, allowing them to excel in imitating human operators' demonstrations. The second limitation of this research is the persistent scarcity of comprehensive real-world data and demonstrations for robot training. Despite the teleoperation modes and acquisition of real-world operation data for robot training, the amount of data

acquired to achieve robust robot intelligence is very limited. This is particularly caused by privacy concerns over data from the construction companies that acquire and use these robots. Future research directions include the use of federated learning mechanisms to harness the data from different construction companies and aggregate the operators' demonstrations in a privacy-preserving way to train the robots to automatically conduct complex construction tasks.

Conclusions

Responding to the call for transformation in the labor-intensive, low-productive, and dangerous construction industry, this research proposed a generalizable framework to accelerate the training of construction robots from human supervision and demonstration in teleoperation mode. This approach aims to foster the adoption and deployment of robots in real construction sites. To this end, this research addressed three technical challenges. First, to address the lack of high-quality training data, a teleoperation architecture was developed. This architecture allows users to control robots to complete construction tasks as an intermediate solution to full automation, while collecting useful human supervision and demonstration data. Teleoperation emerged as a practical means to collect human data for robot training. Second, to reduce a large amount of noise in the collected data for efficient robot training, a keyframe identification and extraction method was proposed to increase the success probability of sampled trajectories. As the importance of each sampled action in the trajectory is not uniformly distributed, a keyframe identification method was proposed. This method can further reduce the sampling rate, helping to reduce the stacked bottlenecks. This method also improves the quality of the expert demonstration. The keyframes of the expert trajectory were found using the proposed RL-based method. The results demonstrated the efficacy of the keyframe methods in sampling the expert trajectories, which can reduce 80% of redundant frames, providing a solid data basis for robot learning. To enable generalizable robot learning for different construction tasks, a hierarchical reinforcement learning structure was proposed. This structure trains model-free policies to accomplish the trajectory tasks by incorporating the extracted keyframe methods, as the keyframe probability was used as an additional reward and was incorporated in the environmental

reward feedback. Third, to bring the extensibility to a wider range of tasks and reduce the computational burden of reinforcement learning training process, an integrated approach that combines vision-based trajectory generation with latent space exploration was proposed. The results illustrate that latent space with the robot action binding outperforms the state-of-the-art methods by 20% improvement in success rate for excavation tasks. The reason is that our proposed method integrated latent space containing the dimension reduced information which can be more accurate and reduced the load of the consecutive model. The proposed robot learning method was demonstrated in excavation experiments for validation. Our method has superior performance as compared to the state-of-the-art and has significant potential for application in construction robots.

Data Availability Statement

Some or all data, models, or code that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgements

This research was funded by the National Science Foundation (NSF) via Grant 2129003 and 2222810. The authors gratefully acknowledge NSF's support. Any opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of NSF and The University of Tennessee, Knoxville.

References

Abdolmaleki, A., J. T. Springenberg, Y. Tassa, R. Munos, N. Heess, and M. Riedmiller. 2018. "Maximum a Posteriori Policy Optimisation." *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR.

Andrew Reynolds. 2022. "CHALLENGES AND OPPORTUNITIES IN THE POST COVID-19 WORLD." *Rider Levett Bucknall*. Accessed February 16, 2023. https://www.rlb.com/oceania/insight/challenges-and-opportunities-in-the-post-covid-19-world/.

Apolinarska, A. A., M. Pacher, H. Li, N. Cote, R. Pastrana, F. Gramazio, and M. Kohler. 2021. "Robotic assembly of timber joints using reinforcement learning." *Autom Constr*, 125: 103569. https://doi.org/10.1016/j.autcon.2021.103569.

Arora, S., and P. Doshi. 2018. "A Survey of Inverse Reinforcement Learning: Challenges, Methods and Progress."

Belousov, B., B. Wibranek, J. Schneider, T. Schneider, G. Chalvatzaki, J. Peters, and O. Tessmann. 2022. "Robotic architectural assembly with tactile skills: Simulation and optimization." *Autom Constr*, 133: 104006. https://doi.org/10.1016/j.autcon.2021.104006.

Bharadhwaj, H., A. Garg, and F. Shkurti. 2020. "LEAF: Latent Exploration Along the Frontier." *Proc IEEE Int Conf Robot Autom*, 2021-May: 677–684. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ICRA48506.2021.9560922.

Chane-Sane, E., C. Schmid, and I. Laptev. 2021. "Goal-Conditioned Reinforcement Learning with Imagined Subgoals." *Proc Mach Learn Res*, 139: 1430–1440. ML Research Press.

Chen, X., and J. D. Storey. 2015. "Consistent Estimation of Low-Dimensional Latent Structure in High-Dimensional Data."

Codevilla, F., E. Santana, A. M. López, and A. Gaidon. 2019. "Exploring the Limitations of Behavior Cloning for Autonomous Driving."

David, O., F.-X. Russotto, M. Da Silva Simoes, and Y. Measson. 2014. "Collision avoidance, virtual guides and advanced supervisory control teleoperation techniques for high-tech construction: framework design." *Autom Constr*, 44: 63–72. https://doi.org/10.1016/j.autcon.2014.03.020.

Ding, Y., C. Florensa, M. Phielipp, and P. Abbeel. 2019. "Goal-conditioned Imitation Learning." *Adv Neural Inf Process Syst*, 32. Neural information processing systems foundation.

Duan, K., and Z. Zou. 2023. "Learning from demonstrations: An intuitive VR environment for imitation learning of construction robots."

Falanga, D., S. Kim, and D. Scaramuzza. 2019. "How Fast Is Too Fast? The Role of Perception Latency in High-Speed Sense and Avoid." *IEEE Robot Autom Lett*, 4 (2): 1884–1891. https://doi.org/10.1109/LRA.2019.2898117.

Fang, B., S. Jia, D. Guo, M. Xu, S. Wen, and F. Sun. 2019. "Survey of imitation learning for robotic manipulation." *Int J Intell Robot Appl*, 3 (4): 362–369. Springer. https://doi.org/10.1007/S41315-019-00103-5.

Fang, K., Y. Zhu, A. Garg, A. Kurenkov, V. Mehta, L. Fei-Fei, and S. Savarese. 2018. "Learning Task-Oriented Grasping for Tool Manipulation from Simulated Self-Supervision." *International Journal of Robotics Research*, 39 (2–3): 202–216. SAGE Publications Inc. https://doi.org/10.1177/0278364919872545.

Finn, C., X. Y. Tan, Y. Duan, T. Darrell, S. Levine, and P. Abbeel. 2015. "Deep Spatial Autoencoders for Visuomotor Learning." *Proc IEEE Int Conf Robot Autom*, 2016-June: 512–519. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ICRA.2016.7487173.

Fujimoto, S., H. Hoof, D. M.-I. conference on, and undefined 2018. 2018. “Addressing function approximation error in actor-critic methods.” *proceedings.mlr.pressS Fujimoto, H Hoof, D MegerInternational conference on machine learning, 2018•proceedings.mlr.press*.

Gambao, E., C. Balaguer, and F. Gebhart. 2000. “Robot assembly system for computer-integrated construction.” *Autom Constr*, 9 (5–6): 479–487. Elsevier. https://doi.org/10.1016/S0926-5805(00)00059-5.

Haarnoja, T., A. Zhou, P. Abbeel, and S. Levine. 2018. “Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor.” *35th International Conference on Machine Learning, ICML 2018*, 5: 2976–2989. International Machine Learning Society (IMLS).

Hartmann, V. N., A. Orthey, D. Driess, O. S. Oguz, and M. Toussaint. 2021. “Long-Horizon Multi-Robot Rearrangement Planning for Construction Assembly.” *IEEE Transactions on Robotics*, 39 (1): 239–252. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/TRO.2022.3198020.

Hasunuma, H., M. Kobayashi, H. Moriyama, T. Itoko, Y. Yanagihara, T. Ueno, K. Ohya, and K. Yokoi. 2002. “A tele-operated humanoid robot drives a lift truck.” *Proceedings-IEEE International Conference on Robotics and Automation*, 3: 2246–2252. https://doi.org/10.1109/ROBOT.2002.1013566.

Hasunuma, H., K. Nakashima, M. Kobayashi, F. Mifune, Y. Yanagihara, T. Ueno, K. Ohya, and K. Yokoi. 2003. “A tele-operated humanoid robot drives a backhoe.” *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, 2998–3004. IEEE.

Hentout, A., M. Aouache, A. Maoudj, I. A.-A. Robotics, and undefined 2019. 2019. “Human–robot interaction in industrial collaborative robotics: a literature review of the decade 2008–2017.” *Taylor & FrancisA Hentout, M Aouache, A Maoudj, I AkliAdvanced Robotics, 2019•Taylor & Francis*, 33 (15–16): 764–799. Robotics Society of Japan. https://doi.org/10.1080/01691864.2019.1636714.

Hester, T., T. Schaul, A. Sendonaris, M. Vecerik, B. Piot, I. Osband, O. Pietquin, D. Horgan, G. Dulac-Arnold, M. Lanctot, J. Quan, J. Agapiou, J. Z. Leibo, and A. Gruslys. 2017a. “Deep Q-learning from Demonstrations.” *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, 3223–3230. AAAI press. https://doi.org/10.1609/aaai.v32i1.11757.

Hester, T., M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, A. Sendonaris, G. Dulac-Arnold, I. Osband, J. Agapiou, J. Leibo, and A. Gruslys. 2017b. “Learning from Demonstrations for Real World Reinforcement Learning.”

Higgins, I., L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, A. Lerchner, and G. Deepmind. 2016. “beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework.”

Ho, J., and S. Ermon. 2016. “Generative Adversarial Imitation Learning.” *Adv Neural Inf Process Syst*, 4572–4580. Neural information processing systems foundation.

Huang, L., Z. Zhu, and Z. Zou. 2023. “To imitate or not to imitate: Boosting reinforcement learning-based construction robotic control for long-horizon tasks using virtual demonstrations.” *Autom Constr*, 146: 104691. https://doi.org/10.1016/j.autcon.2022.104691.

Jayaraman, D., F. Ebert, A. Efros, and S. Levine. 2018. “Time-Agnostic Prediction: Predicting Predictable Video Frames.” *7th International Conference on Learning Representations, ICLR 2019*. International Conference on Learning Representations, ICLR.

Jing, G., T. Tosun, M. Yim, and H. Kress-Gazit. 2018. “Accomplishing high-level tasks with modular robots.” *Auton Robots*, 42 (7): 1337–1354. Springer New York LLC. https://doi.org/10.1007/S10514-018-9738-1.

Kabir, A. M., A. Kanyuck, R. K. Malhan, A. V. Shembekar, S. Thakar, B. C. Shah, and S. K. Gupta. 2019. “Generation of synchronized configuration space trajectories of multi-robot systems.” *Proc IEEE Int Conf Robot Autom*, 2019-May: 8683–8690. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ICRA.2019.8794275.

Kabir, A. M., S. Thakar, P. M. Bhatt, R. K. Malhan, P. Rajendran, B. C. Shah, and S. K. Gupta. 2020. “Incorporating Motion Planning Feasibility Considerations during Task-Agent Assignment to Perform Complex Tasks Using Mobile Manipulators.” *Proc IEEE Int Conf Robot Autom*, 5663–5670. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ICRA40945.2020.9196667.

Kadane, J. B. 2023. “Two Kadane Algorithms for the Maximum Sum Subarray Problem.” *Algorithms*, 16 (11): 519. https://doi.org/10.3390/a16110519.

Keating, S., and N. Oxman. 2013. “Compound fabrication: A multi-functional robotic platform for digital design and fabrication.” *Robot Comput Integr Manuf*, 29 (6): 439–448. https://doi.org/10.1016/j.rcim.2013.05.001.

Kim, D., J. Kim, K. Lee, C. Park, J. Song, and D. Kang. 2009. “Excavator tele-operation system using a human arm.” *Autom Constr*, 18 (2): 173–182. https://doi.org/10.1016/j.autcon.2008.07.002.

Lee, D., and M. Kim. 2021. “Autonomous construction hoist system based on deep reinforcement learning in high-rise building construction.” *Autom Constr*, 128: 103737. https://doi.org/10.1016/j.autcon.2021.103737.

Lenz, I., H. Lee, and A. Saxena. 2013. “Deep Learning for Detecting Robotic Grasps.” *International Journal of Robotics Research*, 34 (4–5): 705–724. SAGE Publications Inc. https://doi.org/10.1177/0278364914549607.

Levine, S., C. Finn, T. Darrell, and P. Abbeel. 2015. “End-to-End Training of Deep Visuomotor Policies.” *Journal of Machine Learning Research*, 17: 1–40. Microtome Publishing.

Levine, S., P. Pastor, A. Krizhevsky, and D. Quillen. 2016. “Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection.” *Learning*, 3 (9).

Li, R., and Z. Zou. 2023. “Enhancing construction robot learning for collaborative and long-horizon tasks using generative adversarial imitation learning.” *Advanced Engineering Informatics*, 58: 102140. Elsevier. https://doi.org/10.1016/J.AEI.2023.102140.

Liang, C.-J., V. Kamat, and C. Menassa. 2019. “Teaching Robots to Perform Construction Tasks via Learning from Demonstration.”

Liang, C.-J., V. R. Kamat, and C. C. Menassa. 2020. “Teaching robots to perform quasi-repetitive construction tasks through human demonstration.” *Autom Constr*, 120: 103370. https://doi.org/10.1016/j.autcon.2020.103370.

Lillicrap, T. P., J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. 2016. “Continuous control with deep reinforcement learning.” *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR.

Liu, D., Z. Wang, B. Lu, M. Cong, H. Yu, and Q. Zou. 2020. “A Reinforcement Learning-Based Framework for Robot Manipulation Skill Acquisition.” *IEEE Access*, 8: 108429–108437. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ACCESS.2020.3001130.

Liu, Y., A. Gupta, P. Abbeel, and S. Levine. 2018. “Imitation from Observation: Learning to Imitate Behaviors from Raw Video via Context Translation.” *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 1118–1125. IEEE.

Liu, Y., M. Habibnezhad, and H. Jebelli. 2021. “Brain-computer interface for hands-free teleoperation of construction robots.” *Autom Constr*, 123: 103523. https://doi.org/10.1016/j.autcon.2020.103523.

Lublasser, E., T. Adams, A. Vollpracht, and S. Brell-Cokcan. 2018. “Robotic application of foam concrete onto bare wall elements - Analysis, concept and robotic experiments.” *Autom Constr*, 89: 299–306. https://doi.org/10.1016/j.autcon.2018.02.005.

Luck, J. P., P. L. McDermott, L. Allender, and D. C. Russell. 2006. “An investigation of real world control of robotic assets under communication latency.” *HRI 2006: Proceedings of the 2006 ACM Conference on Human-Robot Interaction*, 2006: 202–209. Association for Computing Machinery. https://doi.org/10.1145/1121241.1121277.

Luo, Y., K. Dong, L. Zhao, Z. Sun, E. Cheng, H. Kan, C. Zhou, and B. Song. 2021. “Calibration-Free Monocular Vision-Based Robot Manipulations with Occlusion Awareness.” *IEEE Access*, 9: 85265–85276. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ACCESS.2021.3082947.

Mandlekar, A., F. Ramos, B. Boots, S. Savarese, L. Fei-Fei, A. Garg, and D. Fox. 2019. “IRIS: Implicit Reinforcement without Interaction at Scale for Learning Control from Offline Robot Manipulation Data.” *Proc IEEE Int Conf Robot Autom*, 4414–4420. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ICRA40945.2020.9196935.

Mnih, V., K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller. 2013. “Playing Atari with Deep Reinforcement Learning.”

Nasiriany, S., V. H. Pong, S. Lin, and S. Levine. 2019. “Planning with Goal-Conditioned Policies.” *Adv Neural Inf Process Syst*, 32. Neural information processing systems foundation.

Occupational Safety and Health Administration. 2018. “Commonly Used Statistics.” *OSHA Data and Statistics*. Accessed February 22, 2020. https://www.osha.gov/oshstats/commonstats.html.

Pfeiffer, M., S. Shukla, M. Turchetta, C. Cadena, A. Krause, R. Siegwart, and J. Nieto. 2018. “Reinforced Imitation: Sample Efficient Deep Reinforcement Learning for Mapless Navigation by Leveraging Prior Demonstrations.” *IEEE Robot Autom Lett*, 3 (4): 4423–4430. https://doi.org/10.1109/LRA.2018.2869644.

Pore, A., E. Tagliabue, M. Piccinelli, D. Dall'Alba, A. Casals, and P. Fiorini. 2021. "Learning from Demonstrations for Autonomous Soft-tissue Retraction." *2021 International Symposium on Medical Robotics (ISMR)*, 1–7. IEEE.

Praveena, P., G. Subramani, B. Mutlu, and M. Gleicher. 2019. "Characterizing Input Methods for Human-to-Robot Demonstrations." *ACM/IEEE International Conference on Human-Robot Interaction*, 2019-March: 344–353. IEEE Computer Society. https://doi.org/10.1109/HRI.2019.8673310.

Saidi, K. S., T. Bock, and C. Georgoulas. 2016. "Robotics in Construction." *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, eds., 1493–1520. Cham: Springer International Publishing.

Schulman, J., F. Wolski, P. Dhariwal, A. Radford, and O. K. Openai. 2017. "Proximal Policy Optimization Algorithms."

Song, S., A. Zeng, J. Lee, and T. Funkhouser. 2019. "Grasping in the Wild:Learning 6DoF Closed-Loop Grasping from Low-Cost Demonstrations." *IEEE Robot Autom Lett*, 5 (3): 4978–4985. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/LRA.2020.3004787.

Sriram, C., M. Azam, and Mark van Nieuwland. 2015. "The construction productivity imperative." *(accessed February 14, 2023)*. Accessed February 13, 2023. https://www.mckinsey.com/capabilities/operations/our-insights/the-construction-productivity-imperative.

Thakar, S., L. Fang, B. Shah, and S. Gupta. 2018. "Towards Time-Optimal Trajectory Planning for Pick-and-Transport Operation with a Mobile Manipulator." *IEEE International Conference on Automation Science and Engineering*, 2018-August: 981–987. IEEE Computer Society. https://doi.org/10.1109/COASE.2018.8560446.

The National Institute for Occupational Safety and Health (NIOSH). 2019. "NIOSH Strategic Plan: FYs 2019−2026." Accessed February 22, 2020. https://www.cdc.gov/niosh/about/strategicplan/default.html.

Wang, X., X. S. Dong, S. D. Choi, and J. Dement. 2017. "Work-related musculoskeletal disorders among construction workers in the United States from 1992 to 2014." *Occup Environ Med*, 74 (5): 374. https://doi.org/10.1136/oemed-2016-103943.

Xia, P., F. Xu, Z. Song, S. Li, and J. Du. 2023. "Sensory augmentation for subsea robot teleoperation." *Comput Ind*, 145: 103836. https://doi.org/10.1016/j.compind.2022.103836.

Yokoi, K., K. Nakashima, M. Kobayashi, H. Mihune, H. Hasunuma, Y. Yanagihara, T. Ueno, T. Gokyuu, and K. Endou. 2006. "A tele-operated humanoid operator." *journals.sagepub.comK Yokoi, K Nakashima, M Kobayashi, H Mihune, H Hasunuma, Y Yanagihara, T UenoThe International Journal of Robotics Research, 2006•journals.sagepub.com*, 25 (6): 593–602. https://doi.org/10.1177/0278364906065900.

Yu, S.-N., B.-G. Ryu, S.-J. Lim, C.-J. Kim, M.-K. Kang, and C.-S. Han. 2009. "Feasibility verification of brick-laying robot using manipulation trajectory and the laying pattern optimization." *Autom Constr*, 18 (5): 644–655. https://doi.org/10.1016/j.autcon.2008.12.008.

Zakka, K., A. Zeng, J. Lee, and S. Song. 2019. “Form2Fit: Learning Shape Priors for Generalizable Assembly from Disassembly.” *Proc IEEE Int Conf Robot Autom*, 9404–9410. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ICRA40945.2020.9196733.

Zeng, A., P. Florence, J. Tompson, S. Welker, J. Chien, M. Attarian, T. Armstrong, I. Krasin, D. Duong, V. Sindhwani, and J. Lee. 2020. “Transporter Networks: Rearranging the Visual World for Robotic Manipulation.” *Proc Mach Learn Res*, 155: 726–747. ML Research Press.

Zeng, C., H. Zhou, W. Ye, and X. Gu. 2022. “iArm: Design an Educational Robotic Arm Kit for Inspiring Students’ Computational Thinking.” *Sensors 2022, Vol. 22, Page 2957*, 22 (8): 2957. Multidisciplinary Digital Publishing Institute. https://doi.org/10.3390/S22082957.

Zhang, T., S. Guo, T. Tan, X. Hu, and F. Chen. 2020. “Generating Adjacency-Constrained Subgoals in Hierarchical Reinforcement Learning.” *Adv Neural Inf Process Syst*, 2020-December. Neural information processing systems foundation.

Zhao, X., and C. C. Cheah. 2023. “BIM-based indoor mobile robot initialization for construction automation using object detection.” *Autom Constr*, 146: 104647. Elsevier. https://doi.org/10.1016/J.AUTCON.2022.104647.

Zhou, H., J. Xiao, H. Kang, X. Wang, W. Au, and C. Chen. 2022. “Learning-Based Slip Detection for Robotic Fruit Grasping and Manipulation under Leaf Interference.” *Sensors 2022, Vol. 22, Page 5483*, 22 (15): 5483. Multidisciplinary Digital Publishing Institute. https://doi.org/10.3390/S22155483.

Zhou, T., P. Xia, Y. Ye, and J. Du. 2023a. “Embodied Robot Teleoperation Based on High-Fidelity Visual-Haptic Simulator: Pipe-Fitting Example.” *J Constr Eng Manag*, 149 (12): 04023129. American Society of Civil Engineers (ASCE). https://doi.org/10.1061/JCEMD4.COENG-13916/SUPPL_FILE/SUPPLEMENTAL_MATERIALS_JCEMD4.COENG-13916_ZHOU.PDF.

Zhou, T., Q. Zhu, Y. Ye, and J. Du. 2023b. “Humanlike Inverse Kinematics for Improved Spatial Awareness in Construction Robot Teleoperation: Design and Experiment.” *J Constr Eng Manag*, 149 (7): 04023044. American Society of Civil Engineers (ASCE). https://doi.org/10.1061/JCEMD4.COENG-13350/ASSET/0DBE5A52-63E5-4F28-AC68-7E62B533A181/ASSETS/IMAGES/LARGE/FIGURE9.JPG.

Zhu, Y., J. Tremblay, S. Birchfield, and Y. Zhu. 2020. “Hierarchical Planning for Long-Horizon Manipulation with Geometric and Symbolic Scene Graphs.” *Proc IEEE Int Conf Robot Autom*, 2021-May: 6541–6548. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ICRA48506.2021.9561548.

Figure Captions

Figure 1. Structure of our method.

Figure 2. Policy learning architecture.

Figure 3. Extracted keyframes. The Fig. a shows an excavation task example, while the Fig. b displays the collected trajectory in 3D coordinates. Blue dots represent actual robot states, green dots represent human

963 teleoperated control inputs, and red dots signify extracted keyframes. Green dots appear sporadically due
964 to the nature of human teleoperation, where new commands are issued before the previous one is completed.

965 Figure 4. Latent space goal generation network.

966 Figure 5. The loss of goal generation of excavation tasks.

967 Figure 6. Excavation task system setup.

968 Figure 7. Gravity axis alignment.

969 Figure 8. Task success rate. The figure 8.a illustrates our algorithm without latent space representation,
970 where the goal and subgoal states are manually designed and not represented by images. In contrast, the
971 figure 8.b depicts the success rate throughout the training process using latent space representation,
972 eliminating the need for manual intervention. In the 8.a, the blue line stands for the behavior cloning
973 method minus -0.01 to distinguish it from the Gail method success rate.

974 Figure 9. Robot end-effector position estimation.

975 Figure 10. Latent space dimension study for excavation task. (a/b) Train/Validation of excavation task
976 without binding model; (c/d) Train/Validation of excavation task with binding model.