Improving Generalization Ability of Robotic Imitation Learning by Resolving Causal Confusion in Observations

Yifei Chen* **Yuzhe Zhang** **Giovanni Durso** **Nicholas Lawrance** **Brendan Tidd**

Abstract

Recent developments in imitation learning have considerably advanced robotic manipulation. However, current techniques in imitation learning can suffer from poor generalization, limiting performance even under relatively minor domain shifts. In this work, we aim to enhance the generalization capabilities of complex imitation learning algorithms to handle unpredictable changes from the training environments to deployment environments. To avoid confusion caused by observations that are not relevant to the target task, we propose to explicitly learn the causal relationship between observation components and expert actions, employing a framework similar to [6], where a causal structural function is learned by intervention on the imitation learning policy. Disentangling the feature representation from image input as in [6] is hard to satisfy in complex imitation learning process in robotic manipulation, we theoretically clarify that this requirement is not necessary in causal relationship learning. Therefore, we propose a simple causal structure learning framework that can be easily embedded in recent imitation learning architectures, such as the Action Chunking Transformer [29]. We demonstrate our approach using a simulation of the ALOHA [29] bimanual robot arms in Mujoco, and show that the method can considerably mitigate the generalization problem of existing complex imitation learning algorithms.

1 Introduction

Imitation learning (IL) is a powerful framework for robotics, particularly manipulation. This technique allows agents to learn complex skills directly from expert demonstrations without explicitly defining the task or reward [15]. Transformer-based methods, such as the Action Chunking Transformer (ACT) [29] have shown promise due to their ability to handle high-dimensional problems and model temporal dependencies, enabling the completion of a diverse range of tasks. Despite this advancement, their lack of robustness under distribution shift of task or environment impacts their real-world applicability.

One key source of this weak generalization is *causal confusion*, where a model learns spurious correlations between what caused an effect and the result of the effect on the environment [17, 18]. This manifests itself as a mapping between task-relevant *causal* features and irrelevant features, such as focusing on the color of the background or other unimportant objects in the scene. An example is a model that learns to classify hair color and only focuses on gender as a predictive feature [8]. Unfortunately, if the model is tested in a new environment without these irrelevant features and repeats the same task the performance can degrade sharply due to this misaligned learning. Prior work by [6] attempted to solve this problem using causal structure learning, but requires a strong assumption that all observations in the environment are disentangled, which limits its use in robotics.

*All authors are with DATA61 of CSIRO, Australia.
Email addresses: {Yifei.Chen, Yuzhe.Zhang, Giovanni.Durso, Nicholas.Lawrance, Brendan.Tidd}@data61.csiro.au

Preprint. Under review.

In this work, we propose **Causal-ACT**, an extension to IL models that integrates causal structure learning into a transformer-based policy model. Our method avoids the reliance on disentangled representations and learns a causal structure model from an off-the-shelf convolutional encoder, such as ResNet-18 [22]. Our method jointly optimizes over a space of candidate graphs and intervenes to find the appropriate structure that learns the task-relevant features and avoids causal confusion. This allows the policy to focus on what matters and improves its robustness to domain shifts at test time.

We demonstrate our algorithm's effectiveness in simulated experiments using the ALOHA bimanual robot to complete a manipulation task. For these experiments we train the policies with task-irrelevant features in the environment then test in an out-of-distribution environment with the irrelevant features removed. Our results show that our Causal-ACT method substantially outperforms the traditional ACT method under distribution shift conditions. Additionally, we demonstrate that the approach has similar performance to ACT with domain randomization without the need for additional expert demonstrations or computational requirements.

The contributions of this work are: We identify and address the causal confusion problem in Imitation Learning. We provide a theoretical justification for relaxing the disentangled requirement and prove under which graphical conditions you can learn causal structural functions for imitation learning. We develop Causal-ACT, an example of using causal learning in imitation learning without needing the restriction of disentangled embeddings. We empirically prove Causal-ACTS performance through simulation and demonstrate its ability to improve generalization[2].

2 Related work

Imitation learning for manipulation tasks. Imitation learning is often used for tasks in which defining the reward function explicitly is difficult to define or may even have subjective qualities [15, 7]. If there are expert demonstrations available, they can be used to guide the learning of policies to complete the task directly. Transformer-based models [20, 5, 21] have been shown to perform well in IL tasks; performance in robotic manipulation tasks can be further improved through the use of 'action chunking tasks' [29, 3]. Although these methods can deal with some environmental mismatch between training and deployment, they are still vulnerable to large enough domain discrepancies [26].

Methods for Improving Generalization in Robotics. To improve generalization, one dominant strategy is large-scale data collection from diverse real-world environments [30, 16], but this is often impractical due to high costs in time, labor, and computation. An alternative approach to large-scale real-world data collection is using simulated environments to augment or replace real-world data. domain randomization (DR) [23, 13, 1] and generative simulation methods [9] attempt to expose models to various conditions. In particular, DR varies parameters such as lightning, object positions, textures, and friction. However, the quality of the learned models strongly depends on the chosen tuning parameters, quality and quantity of the simulated data [19], costs additional computational resources [14], and can result in overly conservative policies [31]. This results in methods that are fragile in different ways and are computationally expensive to train due to their sample inefficiency.

Causal Methods for Robust Learning. Causal methods have been proposed as a technique to improve the robustness and generalization ability of learning-based methods by modeling the causal relationships between observations and actions [17, 18]. This helps distinguish task-relevant features from spurious correlations, reducing overfitting to irrelevant training patterns and increasing sample efficiency [25]. As a result, models become more resilient to domain shifts. These approaches have shown promise in visual prediction [27], reinforcement learning [28], and imitation learning [6], offering a structured alternative to data-heavy strategies.
The closest related work to our approach is the causal confusion framework proposed by de Haan et al. [6], which aims to identify causally relevant features from disentangled latent representations learned by a Variational Autoencoder (β-VAE). However, this introduces inductive biases [12] and is non-trivial to learn for complex robotic manipulation settings with high-dimensional observations and large, coupled action spaces. In contrast, our work removes the need for disentanglement, which enables improved generalization to out-of-distribution scenarios without relying on extensive data collection.

[2]We defer the theoretical proof, and additional experiment details to the Appendix. Our code can be found in this anonymous repo: https://anonymous.4open.science/r/Causal_ACT_code-E7BA/README.md.

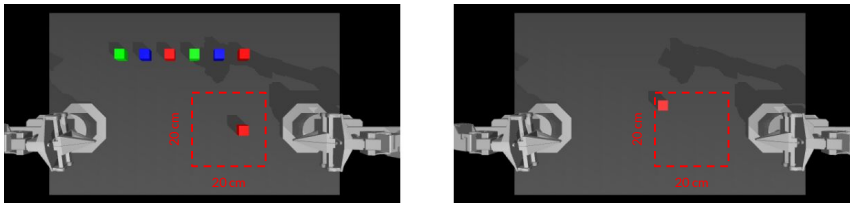

Figure 1: A visualization of the training (left) and the testing (right) environment. Note that the red squares and texts on both figures only indicate the extent of the sampling range for the goal cube initial position and are not included in observations.

3 Insufficient generalizability of imitation learning policies

Imitation learning In imitation learning, a robot tries to learn a *policy*, i.e., a function $\pi : \mathcal{X} \to \mathcal{A}$ taking an *observation* $\mathbf{X}^t$ from the observation space $\mathcal{X}$ and outputting an *action* A^t from the corresponding space $\mathcal{A}$ for each time step t, so that executing the policy generates a temporal sequence of observations and actions that is as close to a set of given *demonstrations* as possible. We assume that an observation $\mathbf{X}^t = (X_1^t, \cdots, X_n^t)$ is an n-dimensional vector, and it can be camera images or their embedded representation (the output of an image encoding network such as a ResNet-50 penultimate layer).

Note that in the above definition, we assume that at each time step, the policy π predicts a single-step action, i.e., $A^t = \pi(\mathbf{X}^t)$. However, π's output can be a sequence of consecutive actions, e.g., in the ACT algorithm [29].

A simple example of performance drop facing domain discrepancy To concretely demonstrate our motivation, we take ACT [29] as an example to illustrate its vulnerability facing a domain discrepancy from the training environment to the test environment. Consider the same task described in our experiment (Section 6), that is, two end-effectors of an ALOHA robot are trained to grasp the target cube (the red one initially located in the red square) with one arm and transfer it to the other. However, from the training environment to the test environment (i.e., from left to right of Figure 1), we remove the six distracting cubes to model an unpredictable change in deployment environment. We observe a drastic performance drop when the robot faces such a mild environment change.

Without proper handling, the robot learns to make decisions based not only on the target, but also on the surroundings. However, the surrounding environment should not be a factor in deciding the next action; instead, the shape and location of the target cube are the real factors. Relying on such task-irrelevant information can obstruct the policy from properly responding to changes in the surroundings. Therefore, it is crucial for the robot to learn which observations decide the next action from the demonstration sequences such that it can learn generalizable skill knowledge instead of a mapping that overfits to the training environment.

4 Resolving causal confusion enhances generalizability

4.1 Modeling the imitation policy as a causal model

Causal graph We model the causal interactions between variables in imitation policies, i.e., observations and actions, in a graphic model. Figure 2 shows an example of such causal relationships. We use a *directed graph* to model the interactions between variables (i.e., observation dimensions and actions) involved in the policy, and we call such a directed graph a *causal graph* [17]. In a causal graph, the nodes are random variables, e.g., observations and actions in Figure 2, and a directed edge denotes a direct causal relationship between variables, e.g., $V_1 \to V_2$ indicates that the change of variable V_1's value directly causes the change of V_2's value, but not vice versa.

In a causal graph $\mathcal{G}$, between two variables V_1 and V_2, if $V_1 \to V_2$, V_1 is a *parent* of V_2, and we denote all parents of V_2 in as $pa_{\mathcal{G}}(V_2)$. A sequence of nodes $(V_1, V_2, \cdots, V_m)$ is called a directed path if $V_i \to V_{i+1}$ for all $1 \leq i \leq m-1$, and any other nodes on the directed path is called a *descendant* of V_1, and we denote V_1's descendants as $de_{\mathcal{G}}(V_1)$, and the other nodes (i.e., non-descendants of V_1)

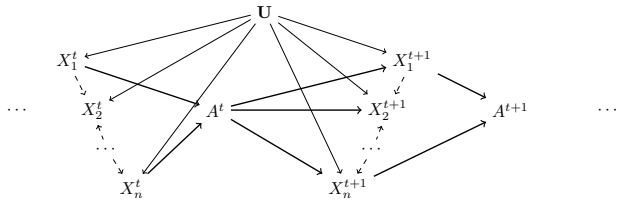

Figure 2: An example causal graph of an imitation policy. At each time step t, the policy π selects an action A^t based on a subset of observations in $\mathbf{X}^t$. The resulting process consequently triggers new observations $\mathbf{X}^{t+1}$ together with the unmeasurable (hidden) exogenous variables $\mathbf{U}$. There may also exist (non-temporal) causal relationships within the observation space (an entangled representation), indicated by dashed arrows between elements in $\mathbf{X}$ (here, X_2 is partially caused by X_1). Causal relationships between the observation states and the policy are shown with solid arrows, so in this case $\mathbf{X}_2$ is part of the observation space but does not have a causal affect on the policy. The goal of this work is to learn these causal relationships from data.

as $nd_{\mathcal{G}}(V_1)$. A causal graph induces the *local Markov property* of involved variables, i.e., for each variable V in the causal graph:

$$V \perp nd_{\mathcal{G}}(V) \mid pa_{\mathcal{G}}(V). \tag{1}$$

That is, V is statistically independent from all its non-descendants conditioned on V's parents.

Shaping an imitation policy π as an above graphical causal model involving each time steps' observations and action, and unmeasurable exogenous variables as the nodes, we have a specific causal graph structure, e.g., one depicted in Figure 2. Concretely, for each time step t, A^t has only in-degrees from a subset of $\mathbf{X}^t$, since π decides A^t only based on the corresponding observations. That is, each observation dimension with an out-degree to the action has a direct influence on the decision of A^t. Then, the next time step's observations $\mathbf{X}^{t+1}$ are consequently determined by the precursor action A^t together with the exogenous variables $\mathbf{U}$.

Notice that the unmeasurable exogenous variable is not accessible to the agent, and thus there is no edge between $\mathbf{U}$ and A^t. Importantly, in practice, it is not necessary that each dimension of the observation has influence on the agent's decision. For instance, in the example introduced in Section 3, the distracting cubes do not contain information of the task target, and thus the corresponding observation should not influence the decision. Our objective in this work is to learn which dimensions of the observation are direct causes of the agent's decision, to prevent the agent from making biased decision by taking uninformed information.

Notice also that in Figure 2, we allow the existence of causal relationships between dimensions of the observation at a time step[3], denoted as dashed edges. This is different from one of the main claims in [6], which requires disentangled representation of the observation in the learning of the causal model. **In practice, learning causally disentangled representation is challenging [11], and it is almost impossible without introducing biases [12].** Therefore, we theoretically address this problem in Section 4.2, by showing that we are guaranteed to learn a consistent structural function (defined as follows) without the disentangled representation condition.

Structural causal model Considering the causal graph modeling the imitation policy π, e.g., the one in Figure 2, we assume that at each time step, the observation dimensions are fixed, e.g., each dimension denotes the information of the same graphical or semantic component of the observation. Therefore, we restrict our analysis to a single time step, that is, learning the causal relationship between variables of $\mathbf{X}^t \cup \{A^t\}$.

Based on a causal graph, we further assume that a *structural causal model* (SCM) exists. That is, the variables in the causal graph subject to counterfactuals with independent errors, i.e., the variables satisfy a data generation process: for each variable V in the causal graph, its value is decided by a *structural function*:

$$A = f_V(pa_{\mathcal{G}}(V), \epsilon_V), \tag{2}$$

[3]Note that such causal relationships between observation dimensions are not temporal as the relationship between $\mathbf{X}^t$ and A^t. Such non-temporal causal relationship can be revealed via observational data [17].

where the noise term ϵ_V is usually introduced due to the unmeasurable exogenous variables $\mathbf{U}$, and we assume the noise terms are mutually independent for each variable pair. Then, an SCM is a tuple $\mathcal{M} = \langle \mathbf{V}, \mathbf{U}, \mathcal{V}, \mathcal{U}, \mathbf{f}, \mathbb{P}_{\mathcal{U}} \rangle$, where $\mathbf{V} = \mathbf{X}^t \cup \{A^t\}$ is the set of endogenous variables (i.e., observable/measurable variables), $\mathbf{U}$ is the set of (unmeasurable/hidden) exogenous variables, $\mathcal{V}$ is the product of domains of $\{V\}_{V \in \mathbf{V}}$, $\mathcal{U}$ is the product of domains of $\{U\}_{U \in \mathbf{U}}$, $\mathbf{f}$ is the set of structural functions, and $\mathbb{P}_{\mathcal{U}}$ is the product of exogenous distributions. Intuitively, the structural function determines how observational data of the variables is generated. For example, the demonstration sequences are generated based on expert's recognition of the cause structure, and thus this data is said to be *faithful* to the SCM.

Then, formally, in this work, we aim at first specifying the subset of observation dimensions in $\mathbf{X}^t$ that directly causes the decision of A^t (i.e., $pa_{\mathcal{G}}(A^t)$), and then learning the mapping from $pa_{\mathcal{G}}(A^t)$ to the agent action A^t:

$$A^t = \hat{f}_\theta(pa_{\mathcal{G}}(A^t), \epsilon_{A^t}), \tag{3}$$

that consistently estimates the true structural function $A^t = f_{A^t}(pa_{\mathcal{G}}(A^t), \epsilon_{A^t})$, with a series of expert demonstration observations and actions.

4.2 Learnability of structural functions without disentangled representation requirement

[6] also tries to enhance imitation learning algorithms' generalization ability by learning a similar causal structure, however, they claim that a necessary condition for the framework is a disentangled representation of the observation, i.e. a mutually independent relationship between observation dimensions. Apparently, forcing the mutual independence condition yields a simple causal graph containing no cycles, which generates a uniquely distributed dataset. Then, an estimation based on such a dataset in turn identically approximates the underlying structural function. However, when the causal relationships are complex, especially when cycles exist in the causal graph, a single SCM may generate multiple distributions, making it impossible to identify the underlying structural function. However, in the case of imitation learning, due to the specific feature of the causal graph as specified in Section 4.1, we theoretically justify that causal connections, even cycles, among observation dimensions does not obstruct identifying the underlying structural function f_{A^t}.

Formally, we would like the imitation policy causal graph to satisfy certain conditions, such that if we properly estimate a mapping $g_{A^t} : pa_{\mathcal{G}}(A^t) \to A^t$, then this mapping uniquely indicates the underlying structural function, i.e.,

$$g_{A^t} \Longleftrightarrow f_{A^t}.$$

As follows, we formally introduce the function learnability property, and prove that it is satisfied in imitation policy causal graphs even if we discard the disentangled representation condition.

Unique Solvability We use the concept of (unique) solvability [4] to demonstrate that when we learn a mapping between a subset of variables, whether this mapping coincidentally interacts with the full SCM. Note that we use the more restricted definition, i.e., unique solvability, as follows.

Definition 1 (Unique Solvability [4])**.** *An SCM $\mathcal{M} = \langle \mathbf{V}, \mathbf{U}, \mathbf{f}, \mathbb{P}_\xi \rangle$ is uniquely solvable w.r.t. $\mathbf{V}' \subseteq \mathbf{V}$ if there exists a measurable mapping $g_{\mathbf{V}'} : \mathcal{V}_{pa_{\mathcal{G}}(\mathbf{V}')\setminus\mathbf{V}'} \times \mathcal{U}_{pa_{\mathcal{G}}(\mathbf{V}')} \to \mathcal{V}_{pa_{\mathcal{G}}(\mathbf{V}')}$ such that for $\mathbb{P}_{\mathcal{U}}$-almost every $\epsilon \in \mathcal{U}$ and for all $\mathbf{v} \in \mathcal{V}$,*

$$\mathbf{v}_{\mathbf{V}'} = g_{\mathbf{V}'}(\mathbf{v}_{pa_{\mathcal{G}}(\mathbf{V}')\setminus\mathbf{V}'}, \epsilon_{pa_{\mathcal{G}}(\mathbf{V}')}) \Longleftrightarrow \mathbf{v}_{\mathbf{V}'} = f_{\mathbf{V}'}(\mathbf{v}, \epsilon).$$

We refer to the following well recognized definition of disentangled representation to formalize the disentangled representation requirement for subsequent analysis.

Definition 2 (Disentangled representation [2])**.** *Disentangled representation should separate the distinct, independent and informative generative factors of variation in the data. Single latent variables are sensitive to changes in single underlying generative factors, while being relatively invariant to changes in other factors.*

Based on this definition, we formalize the requirement as: at each time step t, for each pair of observation dimensions $X_i^t, X_j^t \in \mathbf{X}^t$, they are mutually independent conditioned on the action at $t-1$, i.e.,

$$X_i^t \perp X_j^t \mid A^{t-1}.$$

However, as follows, we prove that this conditional independence condition is not a necessary requirement for unique solvability.

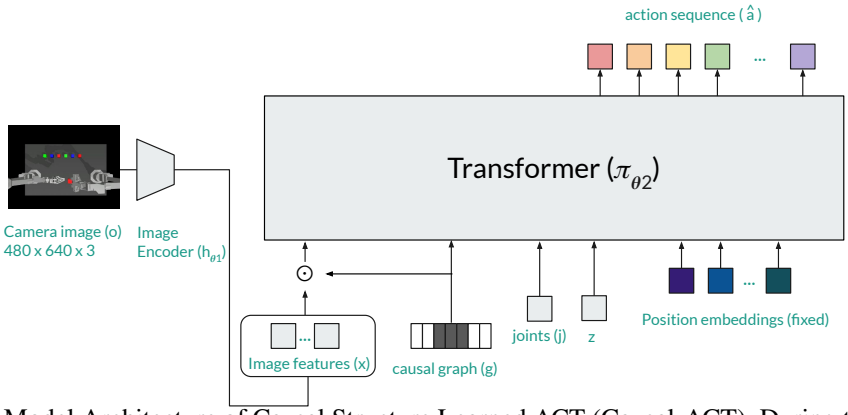

Figure 3: Model Architecture of Causal Structure Learned ACT (Causal-ACT). During training, a causal graph is uniformly sampled to modulate the image features. At test time, the causal graph is fixed to the best-performing graph which is decided by the targeted intervention process. As in ACT, the style variable z is learned by a transformer encoder and set to zero at test time.

Proposition 1. *A SCM* $\mathcal{M} = \langle \mathbf{V}, \mathbf{U}, \mathbf{f}, \mathbb{P}_\xi \rangle$ *corresponding to a imitation policy causal graph is uniquely solvable w.r.t.* $\{A^t\}$.

With relaxing of the disentangled representation requirement, theoretically, we may simply apply the causal structure learning component to a wide range of complex imitation learning algorithms for generalization enhancement. In the next section, we show this simple strategy of incorporating the causal structure learning component in a recent imitation learning algorithm called Action Chunking Transformer (ACT) [29], to the strategy's efficacy.

5 Causal-ACT: Causal Structure Learned ACT

This section introduces our proposed Causal Structure Learned ACT (Causal-ACT). To solve the causal confusion by finding the true causal structure, we intervene the image embedding input of the trained policy to collect interventional data. This strategy is introduced in [6]. First, we train a causal graph-parameterized ACT where various randomly sampled causal graphs are learned jointly with the ACT. Then, we perform the targeted intervention to find the causal graph that is expected to optimize performance.

In our work, Causal-ACT is built based on the ACT [29] by incorporating a causal graph as an additional input and using it to modulate the image features. Figure 3 shows the model architecture of the Causal-ACT. The workflow of Causal-ACT is described in Algorithm 1. Specifically, we use a ResNet encoder h_{θ_1} to extract image features x_t based on the input image observation o_t at timestep t. Then, the image features are multiplied element-wise with a candidate causal graph g_t, which is randomly sampled during training. In particular, the causal graph is represented as an array of binary variables, and the array's dimensionality matches that of the image features. During training, each element of the graph is sampled uniformly i.i.d. As in ACT [29], a style variable z is learned by a transformer encoder $q_\phi(z|a_{t:t+k}, j_t)$, where k is the chunk size of the action sequence and j_t represents joint positions of the robot's arms. The action sequence at time t is predicted based on the graph-parameterized image features $(x_t * g_t)$, the candidate graph (g_t), robot's joints (j_t), and the style variable (z), that is, $\hat{a}_{t:t+k}$ from $\pi_{\theta_2}(\hat{a}_{t:t+k}|x_t * g_t, g_t, j_t, z)$. In each training step, we perform a gradient descent step to minimize the loss function as shown in Equation 4.

$$\mathcal{L} = MSE(\hat{a}_{t:t+k}, a_{t:t+k}) + \beta D_{KL}(q_\phi(z|a_{t:t+k}, j_t))||\mathcal{N}(0, I) \tag{4}$$

where a and $\hat{a}$ represent the demonstrated action and the predicted action, respectively. Note that parameters θ_1 and θ_2 are updated concurrently based on the MSE loss.

After training, we perform the targeted intervention according to Algorithm 1 to search for the graph that performs the best (denoted as g^*) according to the episodic reward. Here, one episode refers to

Algorithm 1 Causal Structure Learned Action Chunking Transformers (Causal-ACT):

Input expert demonstration dataset $\mathcal{D}$, chunk size k, weight β,
and number of training epochs n_epochs
Let a_t, j_t, o_t, and g_t denote the action, robot proprioception, image observation, and sampled causal graph, respectively, at timestep t
Initialize image encoder $h_{\theta_1}(x_t \mid o_t)$, style encoder $q_\phi(z|a_{t:t+k}, j_t)$ and policy $\pi_{\theta_2}(\hat{a}_{t:t+k} \mid x_t * g_t, g_t, j_t, z)$
Initialize $\omega = 0$, $D = \emptyset$
for iteration $n = 1, 2, ..., n\_epochs$ **do**
    Sample o_t, j_t, $a_{t:t+k}$ from $\mathcal{D}$
    Sample z from $q_\phi(z|a_{t:t+k}, j_t)$
    Sample g_t from $\mathcal{U}^{\dim(x)}\{0, 1\}$
    Predict x_t from $h_{\theta_1}(x_t \mid o_t)$
    Predict $\hat{a}_{t:t+k}$ from $\pi_{\theta_2}(\hat{a}_{t:t+k}|x_t * g_t, g_t, j_t, z)$
    $\mathcal{L}_{reconst} = MSE(\hat{a}_{t:t+k}, a_{t:t+k})$
    $\mathcal{L}_{reg} = D_{KL}(q_\phi(z|a_{t:t+k}, j_t))||\mathcal{N}(0, I)$
    Perform a gradient descent step on $\mathcal{L} = \mathcal{L}_{reconst} + \beta\mathcal{L}_{reg}$
    with respect to the network parameters θ_1, θ_2, and ϕ
for $i = 1, ..., N$ **do**
    Sample $g \sim p(g) \propto exp < \omega, g >$
    Compute reward R_g by executing $\pi_{\theta_2}(\hat{a}_{t:t+k}|x_t * g, g, j_t, z)$ with trained image encoder $h_{\theta_1}(x_t \mid o_t)$ and style variable $z = 0$
    $D \leftarrow D \cup \{(g, R_g)\}$
    Fit ω on D with linear regression
return $g^* = \arg\max_g p(g)$, h_{θ_1}, and π_{θ_2}

the entire trajectory of performing the task from the initial to the final timestep. In our experiment, we follow the policy execution intervention method as described in [6], where they use a linear energy-based model to learn the proper causal graph g^*. Intuitively, the probability that a graph is the true graph is proportional to the episodic reward returned by executing the policy based on the graph.

When inferencing Causal-ACT, the causal graph g is set to the best graph g^*, and the style variable z is set to 0. In other words, at inference timestep t, the predicted action sequence of size k is $\hat{a}_{t:t+k}$ = $\pi_{\theta_2}(\hat{a}_{t:t+k}|x_t * g^*, g^*, j_t, 0)$, where $x_t = h_{\theta_1}(x_t \mid o_t)$.

6 Experiments

To empirically evaluate the effectiveness of our proposed method, Causal-ACT, we conduct simulated experiments on a set of bimanual robotic manipulation tasks using the ALOHA platform [29] and MuJoCo simulator [24]. Our experiments were chosen to test generalization under environmental domain shifts, specifically focusing on the impact of causal confusion on performance. We compare our method against two baselines, the original ACT algorithm and domain randomization strategies.

6.1 Experimental Setup

Simulated hardware and software ALOHA consists of two 6-DoF ViperX arms, configured for bimanual manipulation. All experiments are conducted in MuJoCo version 2.3.7 with top-view RGB camera inputs of resolution 480×640 with a control rate of 50 Hz. We simulate manipulation tasks on a laptop running Ubuntu 22.04 with Python 3.8, 64 GB RAM, a 13th Gen Intel Core i7-13850HX CPU, and an NVIDIA RTX 2000 Ada GPU.

Task Description We use a variant of the *Cube Transfer* task [29]. In this task the right arm must pick up a red cube and transfer it to the left arm. The cube's initial location is uniformly randomly sampled within an area of 20cm×20cm, as shown by the red square in Figure 1. The task allows partial completion and an associated reward from 1 to 4 based on whether the arm can *touch* the cube, *lift* it, *attempt to transfer*, and *complete the transfer* from one end effector to the other.

Table 1: Success rates for ACT and Causal-ACT across distribution shifts. Each result is averaged over three seeds and 50 evaluation episodes.

Method	**Out-of-Distribution**			**In-Distribution**		
	Touched	Lifted	Transfer	Touched	Lifted	Transfer
ACT	0.87	0.54	0.23	0.99	0.96	0.89
Causal-ACT	0.88	0.84	0.82	0.96	0.96	0.96

For imitation learning purposes we gather a series of demonstrative 'expert' trajectories using scripted end-effector control with full knowledge of the ground-truth state. Each of these training episodes consists of 400 steps (8 s) and is stored as an HDF5 file (~ 370 MB each). This dataset comprises trajectories that include both joint and image data and is used to train the imitation algorithms.

Training vs Test Distribution To introduce causal confusion, six irrelevant distracting cubes with fixed color and position are added during training as shown in the left panel of Figure 1. Then, to examine the model's robustness, the trained agent is evaluated in an out-of-distribution (OOD) environment where there are no distracting cubes, as shown in the right panel of Figure 1. This simulates a domain shift where the environmental variables may appear to change, but the task remains the same.

Baselines In this work, we compare against two baseline methods, the original ACT algorithm and ACT trained with domain randomization. For the domain randomization method, ACT is trained with demonstrations containing a varying number, position and color of distractors. This setup follows conventional DR setups. We sample a variety of training environments with 1-6 distractors using a power weighted discrete distribution $P(i) = \frac{i^k}{\sum_{j=1}^{k} j^k}$,. Where k controls the skew of the distribution. Larger values of k result in examples with higher numbers of cubes, which are further from the test distribution. Similarly, $k = 0$ yields a uniform sampling of the number of cubes, which is closer to the test environment (see Figure 4 in Appendix C). Because the test environment contains no distractor cubes, environments with fewer cubes are closer to the test distribution.

Training Details Each ACT model has approximately 84M parameters. Training 2000 epochs takes around 55 minutes using 4GB of VRAM. All models are trained on the same expert dataset (unless otherwise stated) with three random seeds. Inference of the models is real-time (50 Hz generating a chunk of 100 steps). The Causal-ACT model is trained for 2000 epochs and takes around 65 minutes using 5GB of VRAM. The hyperparameters of training ACT and Causal-ACT are shown in Appendix B.

7 Results

Generalization Performance Table 1 presents results of ACT and our Causal-ACT methods trained on the distractor training environment. Here we discuss the complete task ('transfer') results unless noted otherwise. While ACT performs well in the in-distribution environment (0.89), its performance degrades substantially when tested out-of-distribution (0.23). This experiment shows an example of causal confusion: the ACT model has learned spurious correlations between the distractors and the task performance. Contrastingly, our Causal-ACT method maintains strong performance in-distribution (0.96) and substantially improves out-of-distribution performance (0.88) up from (0.23). This improved performance is achieved without needing additional expert training data or modeling additional environment variance. It was learned by learning to ignore the irrelevant components of the observation that do not lead to task performance.

Comparison to domain randomization Domain randomization can improve the performance if the sampled training environments overlap with the test environment enough. Our results for various k values are summarized in Table 2.

Domain randomization has a varying level of performance based on how close the randomly sampled training domains are to the test domain. As expected, lower k values (higher likelihood of fewer

Table 2: Success rates of plain ACT (row 1), Domain randomization with different sampling distributions (row 2-5), Causal-ACT (row 6), and ablation study of Causal-ACT with random causal graphs (row 7-8), for Out-of-Distribution tests. Each success rate is averaged over performances of three trained ACT agents that are trained with three different random seeds. Each agent is trained for 2000 epochs and evaluated for 50 episodes.

Method	Success rate		
	Touched	Lifted	Transfer
ACT	0.87	0.54	0.23
ACT + DR ($k = +\infty$)	0.75	0.59	0.34
ACT + DR ($k = 6$)	0.87	0.7	0.48
ACT + DR ($k = 3$)	0.98	0.91	0.61
ACT + DR ($k = 0$)	1.0	0.98	0.91
Causal-ACT	0.88	0.84	0.82
Causal-ACT (random graph)	0.91	0.82	0.48
Causal-ACT (full-connection graph)	0.1	0.08	0.02

distractor cubes) correspond to increased performance (0.91) because the sampled environments are more similar to the test domain. Conversely, the domains with the largest train to test variance $k = \infty$ have similar performance to the models not using domain randomization. While domain randomization can improve performance, it requires extensive sampling and the assumption that the domain configurations and the sampled domains during training are relevant to testing conditions. Furthermore, randomization over irrelevant features can degrade performance.

Causal-ACT achieves a competitive level of performance with the best DR models while using significantly less data, requiring no manual domain design, and avoiding the issue of over-randomization. Our results suggest that causal structure learning is a data-efficient and robust solution to generalization in imitation learning. We believe this is particularly valuable for real-world conditions where collecting additional data with domain randomization can be prohibitively expensive.

Ablation study We conduct an ablation study by sampling a causal graph at random (row 7, Table 1), and by using a fully connected graph between observation dimensions and action (row 8, Table 1). Both experiments generate significantly lower performances (0.48 and 0.02) compared to Causal-ACT (row 7, Table 2), showing the efficiency of Causal-ACT's graph searching process.

It is worth noticing that the success rate of the full-connection graph is considerably low (0.02). We conjecture that Causal-ACT suffers from poor performance when the complete observation space is passed, both due to the high amount of irrelevant data and the lack of training in this regime.

8 Conclusion

In this work, we addressed the problem of causal confusion in Imitation Learning (IL), where models rely on spurious correlations that impact their ability to generalize to new environments. Unlike prior work that depended on disentangled representations, we show that causal structure learning can be integrated directly into IL models. We developed Causal-ACT, a transformer-based IL framework that incorporates causal graph optimization to allow the policy to focus on task-relevant features. Our approach improves generalisation and sample efficiency, outperforming the ACT baseline and domain randomisation. Future work will extend this method to other IL architectures and real-world tasks, and explore alternative causal learning techniques to further enhance generalization.

Limitations We would like to specify two limitations of the work. Due to the general high dimensions of image inputs or their embedded representations, e.g. in our case, the ResNet encodes images into $n = 512 * 15 * 20$ dimensional arrays, it is intractable to visit each possible graph from the large space (with size 2^n). Then, an efficient graph sampling method is desirable to differentiate good options fast, and this is one of our future works.

References

[1] Peter Anderson, Ayush Shrivastava, Joanne Truong, Arjun Majumdar, Devi Parikh, Dhruv Batra, and Stefan Lee. Sim-to-real transfer for vision-and-language navigation. In *Conference on Robot Learning*, pages 671–681. PMLR, 2021.

[2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8): 1798–1828, 2013.

[3] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. $\pi 0$: A vision-language-action flow model for general robot control. *URL https://arxiv. org/abs/2410.24164*, 2024.

[4] Stephan Bongers, Patrick Forré, Jonas Peters, and Joris M Mooij. Foundations of structural causal models with cycles and latent variables. *The Annals of Statistics*, 49(5):2885–2915, 2021.

[5] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control, 2023.

[6] Pim De Haan, Dinesh Jayaraman, and Sergey Levine. Causal confusion in imitation learning. *Advances in neural information processing systems*, 32, 2019.

[7] Jonathan Ho and Stefano Ermon. Generative Adversarial Imitation Learning. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[8] Pavel Izmailov, Polina Kirichenko, Nate Gruver, and Andrew G Wilson. On feature learning in the presence of spurious correlations. *Advances in Neural Information Processing Systems*, 35: 38516–38532, 2022.

[9] Stephen James, Paul Wohlhart, Mrinal Kalakrishnan, Dmitry Kalashnikov, Alex Irpan, Julian Ibarz, Sergey Levine, Raia Hadsell, and Konstantinos Bousmalis. Sim-To-Real via Sim-To-Sim: Data-Efficient Robotic Grasping via Randomized-To-Canonical Adaptation Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[10] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[11] Aneesh Komanduri, Yongkai Wu, Feng Chen, and Xintao Wu. Learning causally disentangled representations via the principle of independent causal mechanisms. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 4308–4316, 2024.

[12] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pages 4114–4124. PMLR, 2019.

[13] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, and Gavriel State. Isaac Gym: High Performance GPU Based Physics Simulation For Robot Learning. *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 1, 2021.

[14] OpenAI, Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, Jonas Schneider, Nikolas Tezak, Jerry Tworek, Peter Welinder, Lilian Weng, Qiming Yuan, Wojciech Zaremba, and Lei Zhang. Solving Rubik's Cube with a Robot Hand, 2019.

[15] Takayuki Osa, Joni Pajarinen, Gerhard Neumann, J. Andrew Bagnell, Pieter Abbeel, and Jan Peters. An Algorithmic Perspective on Imitation Learning. *Foundations and Trends® in Robotics*, 7(1-2), 2018. ISSN 1935-8253, 1935-8261. doi: 10.1561/2300000053.

[16] Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.

[17] Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, New York, 2000.

[18] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

[19] Aayush Prakash, Shaad Boochoon, Mark Brophy, David Acuna, Eric Cameracci, Gavriel State, Omer Shapira, and Stan Birchfield. Structured Domain Randomization: Bridging the Reality Gap by Context-Aware Synthetic Data. In *2019 International Conference on Robotics and Automation (ICRA)*, 2019. doi: 10.1109/ICRA.2019.8794443.

[20] Nur Muhammad Shafiullah, Zichen Cui, Ariuntuya (Arty) Altanzaya, and Lerrel Pinto. Behavior Transformers: Cloning k modes with one stone. *Advances in Neural Information Processing Systems*, 35, 2022.

[21] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Perceiver-Actor: A Multi-Task Transformer for Robotic Manipulation. In *Proceedings of The 6th Conference on Robot Learning*. PMLR, 2023.

[22] Sasha Targ, Diogo Almeida, and Kevin Lyman. Resnet in resnet: Generalizing residual architectures. *arXiv preprint arXiv:1603.08029*, 2016.

[23] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017. doi: 10.1109/IROS.2017.8202133.

[24] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109.

[25] Liuyi Yao, Zhixuan Chu, Sheng Li, Yaliang Li, Jing Gao, and Aidong Zhang. A Survey on Causal Inference. *ACM Trans. Knowl. Discov. Data*, 15(5), 2021. ISSN 1556-4681. doi: 10.1145/3444944.

[26] Maryam Zare, Parham M. Kebria, Abbas Khosravi, and Saeid Nahavandi. A Survey of Imitation Learning: Algorithms, Recent Developments, and Challenges. *IEEE Transactions on Cybernetics*, 54(12), 2024. ISSN 2168-2275. doi: 10.1109/TCYB.2024.3395626.

[27] Andy Zeng, Pete Florence, Jonathan Tompson, Stefan Welker, Jonathan Chien, Maria Attarian, Travis Armstrong, Ivan Krasin, Dan Duong, Vikas Sindhwani, and Johnny Lee. Transporter Networks: Rearranging the Visual World for Robotic Manipulation. In *Proceedings of the 2020 Conference on Robot Learning*. PMLR, 2021.

[28] Amy Zhang, Clare Lyle, Shagun Sodhani, Angelos Filos, Marta Kwiatkowska, Joelle Pineau, Yarin Gal, and Doina Precup. Invariant causal prediction for block mdps. In *International Conference on Machine Learning*, pages 11214–11224. PMLR, 2020.

[29] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems XIX*, Daegu, Republic of Korea, July 2023. doi: 10.15607/RSS.2023.XIX.016.

[30] Tony Z Zhao, Jonathan Tompson, Danny Driess, Pete Florence, Kamyar Ghasemipour, Chelsea Finn, and Ayzaan Wahid. Aloha unleashed: A simple recipe for robot dexterity. *arXiv preprint arXiv:2410.13126*, 2024.

[31] Wenshuai Zhao, Jorge Peña Queralta, and Tomi Westerlund. Sim-to-real transfer in deep reinforcement learning for robotics: a survey. In *2020 IEEE symposium series on computational intelligence (SSCI)*, pages 737–744. IEEE, 2020.

A Proof of Proposition 1

Proof. To prove Proposition 1, we need the following lemma.

Lemma 1 ([4]). *A SCM $\mathcal{M} = \langle \mathbf{V}, \mathbf{U}, \mathbf{f}, \mathbb{P}_\xi \rangle$ is uniquely solvable w.r.t. a single variable $\{V\} \subseteq \mathbf{V}$ if and only if V has no self cycle in the corresponding causal graph, i.e., edge $V \to V$ does not exist.*

According to the specification of a imitation learning causal graph, variable A^t only has in-degrees from a subset of $\mathbf{X}^t$. Then, the causal graph satisfies the condition required in Lemma 1, and therefore the SCM corresponding to the causal graph is uniquely solvable. □

B Hyperparameters of Training

For a fair comparison, we keep the hyperparameters of training ACT and Causal-ACT the same as reported in [29]. As is shown in Table 3, intervention iteration is the extra hyperparameter for Causal-ACT to get a best-performed graph. Both ACT and Causal-ACT are trained for 2000 epochs with Adam [10] of the standard values of $\beta_1 = 0.9$ and $\beta_2 = 0.999$ as the optimizer.

Table 3: Hyperparameters of training ACT and Causal-ACT.

learning rate	10^{-5}
chunk size	100
batch size	8
number of heads	8
number of encoder layers	4
number of decoder layers	7
hidden dimension	512
feedforward dimension	3200
learning rate of backbone	10^{-5}
beta	10
intervention iteration	50

C Domain randomization sampling distributions

Figure 4 illustrates the likelihood of sampling $i = 1, \ldots, 6$ cubes under the power weighted discrete distribution for values of $k \in \{0, 3, 6, \to \infty\}$.

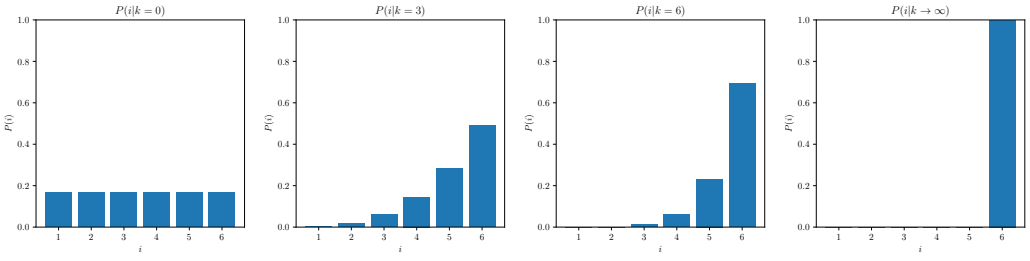

Figure 4: The distribution used to sample the number of cubes in domain randomization.