

BISCUIT: Causal Representation Learning from Binary Interactions

Phillip Lippe¹ Sara Magliacane^{2,3} Sindy Löwe² Yuki M. Asano¹ Taco Cohen⁴ Efstratios Gavves¹

¹QUVA Lab, University of Amsterdam

²AMLab, University of Amsterdam

³MIT-IBM Watson AI Lab

⁴Qualcomm AI Research*, Amsterdam, Netherlands

Abstract

Identifying the causal variables of an environment and how to intervene on them is of core value in applications such as robotics and embodied AI. While an agent can commonly interact with the environment and may implicitly perturb the behavior of some of these causal variables, often the targets it affects remain unknown. In this paper, we show that causal variables can still be identified for many common setups, e.g., additive Gaussian noise models, if the agent’s interactions with a causal variable can be described by an unknown binary variable. This happens when each causal variable has two different mechanisms, e.g., an observational and an interventional one. Using this identifiability result, we propose BISCUIT, a method for simultaneously learning causal variables and their corresponding binary interaction variables. On three robotic-inspired datasets, BISCUIT accurately identifies causal variables and can even be scaled to complex, realistic environments for embodied AI. Project page: philippe.github.io/BISCUIT/.

1 INTRODUCTION

Learning a low-dimensional representation of an environment is a crucial step in many applications, *e.g.*, robotics (Lesort et al., 2018), embodied AI (Kolve et al., 2017) and reinforcement learning (Hafner et al., 2021; Träuble et al., 2022). A promising direction for learning robust and actionable representations is *causal representation learning* (Schölkopf et al., 2021), which aims to identify the underlying causal variables and their relations in a given environment from high-dimensional observations, *e.g.*, images. However, learning causal variables from high-dimensional observations is a considerable challenge and may not always

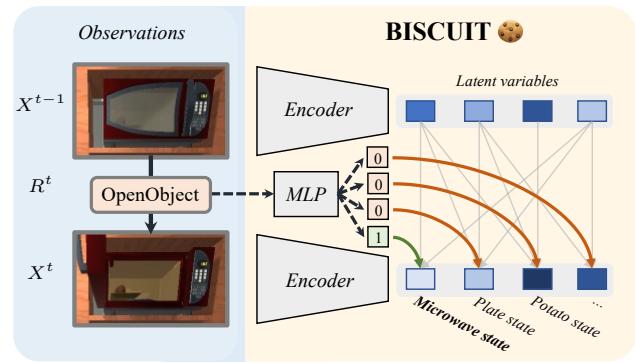


Figure 1: BISCUIT identifies causal variables from images X^{t-1} and X^t , by learning to encode an observable regime variable R^t , *e.g.*, an action, as binary variables. Conditioning each latent on one of these binary variables identifies causal variables in environments like iTHOR (Kolve et al., 2017).

be possible, since multiple underlying causal systems could generate the same data distribution (Hyvärinen et al., 1999). To overcome this, several works make use of additional information, *e.g.*, by using counterfactual observations (Ahuja et al., 2022; Brehmer et al., 2022; Locatello et al., 2020), observed intervention targets (Lippe et al., 2022b, 2023). Alternatively, one can restrict the distributions of causal variables, *e.g.*, by considering environments with non-stationary noise (Khemakhem et al., 2020a; Yao et al., 2022a,b) or sparse causal relations (Lachapelle et al., 2022a,b).

In this paper, instead, we focus on interactive environments, where an agent can perform actions which may have an effect on the underlying causal variables. We will assume that these interactions between the agent and the causal variables can be described by *binary* variables, *i.e.*, that with the agent’s actions, we can switch between two mechanisms, or distributions, of a causal variable, similarly to performing soft interventions. Despite being binary, these interactions include a wide range of common scenarios, such as a robot pressing a button, opening/closing a door, or even colliding with a moving object and alternating its course.

* Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

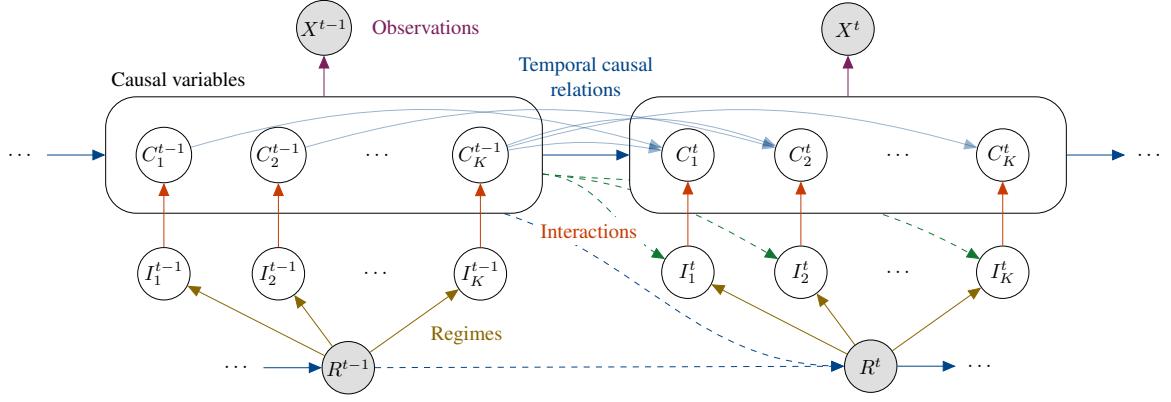


Figure 2: A representation of our assumptions. Observed variables are shown in gray (X^τ and R^τ) and latent variables in white. Optional causal edges are shown as dashed lines. A latent causal variable C_i^t has as parents a subset of the causal factors at the previous time step $C^{t-1} = \{C_1^{t-1}, \dots, C_K^{t-1}\}$, and its latent **binary interaction variable** I_i^t . The interaction variables are determined by an observed **regime variable** R^t and potentially by the variables from the previous time step C^{t-1} (e.g., in a collision). The regime variable can be a dynamical process over time as well, for example, by depending on the previous time step. The **observation** X^τ is a high-dimensional entangled representation of all causal variables C^τ at time step τ .

In this setup, we prove that causal variables are identifiable if the agent interacts with each causal variable in a distinct pattern, *i.e.*, does not always interact with any two causal variables at the same time. We show that for K variables, we can in many cases fulfill this by having as few as $\lfloor \log_2 K \rfloor + 2$ actions with sufficiently diverse effects, allowing identifiability even for a limited number of actions. The binary nature of the interactions permits the identification of a wider class of causal models than previous work in a similar setup, including the common, challenging additive Gaussian noise model (Hyvärinen et al., 1999).

Based on these theoretical results, we propose BISCUIT (**B**inary **I**nteractions for **C**ausal **I**dentifiability). BISCUIT is a variational autoencoder (Kingma et al., 2014) which learns the causal variables and the agent’s binary interactions with them in an unsupervised manner (see Figure 1). In experiments on robotic-inspired datasets, BISCUIT identifies the causal variables and outperforms previous methods. Furthermore, we apply BISCUIT to the realistic 3D embodied AI environment iTHOR (Kolve et al., 2017), and show that BISCUIT is able to generate realistic renderings of unseen causal states in a controlled manner. This highlights the potential of causal representation learning in the challenging task of embodied AI. In summary, our contributions are:

- We show that under mild assumptions, binary interactions with unknown targets identify the causal variables from high-dimensional observations over time.
- We propose BISCUIT, a causal representation learning framework that learns the causal variables and their binary interactions simultaneously.
- We empirically show that BISCUIT identifies both the causal variables and the interaction targets on three robotic-inspired causal representation learning benchmarks, and allows for controllable generations.

2 PRELIMINARIES

In this paper, we consider a causal model \mathcal{M} as visualized in Figure 2. The model \mathcal{M} consists of K latent causal variables C_1, \dots, C_K which interact with each other over time, like in a dynamic Bayesian Network (DBN) (Dean et al., 1989; Murphy, 2002). In other words, at each time step t , we instantiate the causal variables as $C^t = \{C_1^t, \dots, C_K^t\} \in \mathcal{C}$, where $\mathcal{C} \subseteq \mathbb{R}^K$ is the domain. In terms of the causal graph, each variable C_i^t may be caused by a subset of variables in the previous time step $\{C_1^{t-1}, \dots, C_K^{t-1}\}$. For simplicity, we restrict the temporal causal graph to only model dependencies on the previous time step. Yet, as we show in Appendix B.3, our results can be trivially extended to longer dependencies, *e.g.*, $(C^{t-2}, C^{t-1}) \rightarrow C^t$, since C^{t-1} is only used for ensuring conditional independence. As in DBNs, we consider the graph structure to be time-invariant.

Besides the intra-variable dynamics, we assume that the causal system is affected by a regime variable R^t with arbitrary domain \mathcal{R} , which can be continuous or discrete of arbitrary dimensionality. This regime variable can model any known external causes on the system, which, for instance, could be a robotic arm interacting with an environment. For the causal graph, we assume that the effect of the regime variable R^t on a causal variable C_i^t can be described by a latent **binary interaction variable** $I_i^t \in \{0, 1\}$. This can be interpreted as each causal variable having two mechanisms/distributions, *e.g.*, an observational and an interventional mechanism, which has similarly been assumed in previous work (Brehmer et al., 2022; Lippe et al., 2022b, 2023). Thereby, the role of the interaction variable I_i^t is to select the mechanism, *i.e.*, observational or interventional, at time step t . For example, a collision between an agent and an object is an interaction that switches the dynamics of the object from

its natural course to a perturbed one. In this paper, we consider the interaction variable I_i^t to be an unknown function of the regime variable and the previous causal variables, *i.e.*, $I_i^t = f_i(R^t, C^{t-1})$. The dependency on the previous time step allows us to model interactions that only occur in certain states of the system, *e.g.*, a collision between an agent (modeled by R^t) with an object with position C_i^{t-1} will only happen for certain positions of the agent and the object.

We consider the causal graph of Figure 2 to be causally sufficient, *i.e.*, we assume there are no other unobserved confounders except the ones we have described in the previous paragraphs and represented in the Figure, and that the causal variables within the same time step are independent of each other, conditioned on the previous time step and their interaction variables. We summarize the dynamics as $p(C^t|C^{t-1}, R^t) = \prod_{i=1}^K p(C_i^t|C^{t-1}, I_i^t)$. Although C_i^t only depends on a subset of C^{t-1} , w.l.o.g. we model it as depending on all causal variables from the previous time step.

In causal representation learning, the task is to identify causal variables from an entangled, potentially higher-dimensional representation, *e.g.*, an image. We consider an injective observation function g , mapping the causal variables C^t to an observation $X^t = g(C^t)$. Following Klindt et al. (2021); Yao et al. (2022b), we assume g to be defined everywhere for C^t and differentiable almost everywhere. In our setting, once we identify the causal variables, the causal graph can be trivially learned by testing for conditional independence, since the causal graph is limited to edges following the temporal dimension, *i.e.*, from C^{t-1} to C^t . We provide further details on the graph discovery and an example on learned causal variables in Appendix B.4.

3 IDENTIFYING CAUSAL VARIABLES

Our goal in this paper is to identify the causal variables C_1, \dots, C_K of a causal system from sequences of observations (X^t, R^t) . We first define the identifiability class that we consider. We then provide an intuition on how binary interactions enable identifiability, before presenting our two identifiability results. The practical algorithm based on these results, BISCUIT, is presented in Section 4.

3.1 IDENTIFIABILITY CLASS AND DEFINITIONS

Intuitively, we seek to estimate an observation function \hat{g} , which maps a latent space $\hat{\mathcal{C}}$ to observations X , and models each true causal variable C_i in a different dimension of the latent space $\hat{\mathcal{C}}$. This observation function should be equivalent to the true observation function g , up to permuting and transforming the variables individually, *e.g.*, through scaling. Several previous works (Khemakhem et al., 2020a; Lachapelle et al., 2022b; Yao et al., 2022a,b) have considered equivalent identifiability classes, which we define as:

Definition 3.1. Consider a model $\mathcal{M} = \langle g, f, \omega, \mathcal{C} \rangle$ with an injective function $g(C) = X$ with $C \in \mathcal{C}$ and a latent distribution $p_\omega(C^t|C^{t-1}, R^t)$, parameterized by ω and defined:

$$p_\omega(C^t|C^{t-1}, R^t) = \prod_{i=1}^K p_{\omega,i}(C_i^t|C^{t-1}, f_i(R^t, C^{t-1})),$$

where $f_i : \mathcal{R} \times \mathcal{C} \rightarrow \{0, 1\}$ outputs a binary variable for the variable C_i^t . We call \mathcal{M} identifiable iff for any other model $\tilde{\mathcal{M}} = \langle \tilde{g}, \tilde{f}, \tilde{\omega}, \tilde{\mathcal{C}} \rangle$ with the same observational distribution $p(X^t|X^{t-1}, R^t)$, g and \tilde{g} are equivalent up to a component-wise invertible transformation T and a permutation π :

$$p_{\mathcal{M}}(X^t|X^{t-1}, R^t) = p_{\tilde{\mathcal{M}}}(X^t|X^{t-1}, R^t) \Rightarrow g = \tilde{g} \circ T \circ \pi$$

To achieve this identifiability, we rely on the interaction variables I_i^t being binary and having *distinct interaction patterns*, a weaker form of faithfulness on the interaction variables. Intuitively, we do not allow that any two causal variables to have identical interaction variables I_i^t, I_j^t across the whole dataset, *i.e.*, being always interacted with at the same time. Similarly, if all I_i^t are always zero ($\forall t, i: I_i^t = 0$), then we fall back into the well-known unidentifiable setting of non-linear ICA (Hyvärinen et al., 1999). Since interaction variables can also be functions of the previous state, we additionally assume that for all possible previous states, the interaction variables cannot be deterministic functions of any other. Thus, we assume that all causal variables have *distinct interaction patterns*, which we formally define as:

Definition 3.2. A causal variable C_i in $\mathcal{M} = \langle g, f, \omega, \mathcal{C} \rangle$ has a **distinct interaction pattern** if for all values of C^{t-1} , its interaction variable $I_i^t = f_i(R^t, C^{t-1})$ is not a deterministic function $b: \{0, 1\} \rightarrow \{0, 1\}$ of any other I_j^t :

$$\forall C^{t-1}, \forall j \neq i, \nexists b, \forall R^t: f_i(R^t, C^{t-1}) = b(f_j(R^t, C^{t-1})).$$

This assumption generalizes the intervention setup of Lippe et al. (2022c), which has a similar condition on its binary intervention variables, but assumed them to be independent of the previous time step. This implies that we can create a distinct interaction pattern for each of the K causal variables by having as few as $\lfloor \log_2 K \rfloor + 2$ different values for R^t , if the interaction variables are independent of C^{t-1} . In contrast, other methods in similar setups that also exploit an external, temporally independent, observed variable (Khemakhem et al., 2020a; Yao et al., 2022a,b) require the number of regimes to scale linearly with the number of causal variables. If the interaction variables depend on C^{t-1} , the lower bound of the number of different values for R^t depends on the causal model \mathcal{M} , more specifically its interaction functions f_i . Concretely, the lower bound for a causal model \mathcal{M} is the smallest set of values of R^t that ensure different interaction patterns for all C^{t-1} in \mathcal{M} . In the worst case, each C^{t-1} may require different values of R^t to fulfill the condition of Theorem 3.2, such that R^t would need to

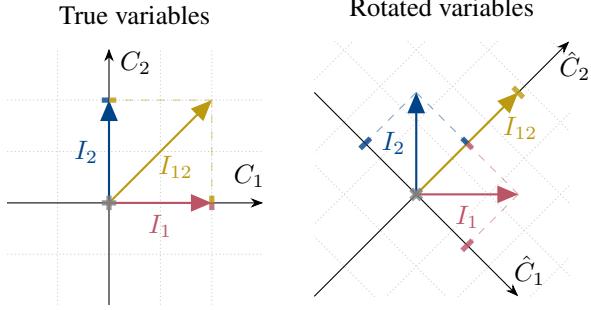


Figure 3: Binary interactions identify the additive Gaussian noise model in Equation (1). The plots show the change of the mean for each variable for a fixed C^{t-1} under interactions affecting only one variable ($I_1 = 1$ or $I_2 = 1$), and under joint interactions I_{12} ($I_1 = I_2 = 1$). Ticks on axes show mean values for each variable, where the mean for $(I_1 = 0, I_2 = 0)$ lies at the origin. The colors of the ticks match the interaction color. **Left:** true causal variables C_1 and C_2 . Each variable has two possible means after any of the possible interactions. The effect of the interactions can be described by a binary variable per axis. **Right:** a rotated representation. Both \hat{C}_1 and \hat{C}_2 have three different means, which cannot be described by a binary variable per axis anymore.

be of the same domain as C^{t-1} (for instance being continuous). At the same time, for models \mathcal{M} in which the condition of Theorem 3.2 can be fulfilled by the same values of R^t for all C^{t-1} , we again recover the lower bound of $\lfloor \log_2 K + 2 \rfloor$ different values of R^t .

3.2 INTUITION: ADDITIVE GAUSSIAN NOISE

We first provide some intuition on how binary interactions, *i.e.*, knowing that each variable has exactly two potential mechanisms, enable identifiability, even when we do not know which variables are interacted with at each time step. We take as an example an additive Gaussian noise model with two variables C_1, C_2 , each described by the equation:

$$C_i^t = \mu_i(C^{t-1}, I_i^t) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2), \quad (1)$$

where ϵ_i is additive noise with variance σ^2 , and μ_i a function for the mean with $\mu_i(C^{t-1}, I_i^t = 0) \neq \mu_i(C^{t-1}, I_i^t = 1)$. Due to the rotational invariance of Gaussians, the true causal variables C_1, C_2 and their rotated counterparts \hat{C}_1, \hat{C}_2 model the same distribution with the same factorization: $\prod_{i=1}^2 p_i(C_i^t | C^{t-1}, R^t) = \prod_{i=1}^2 \hat{p}_i(\hat{C}_i^t | \hat{C}^{t-1}, R^t)$. This property makes the model unidentifiable in many cases (Hyvärinen et al., 2019; Khemakhem et al., 2020a; Lachapelle et al., 2022b; Yao et al., 2022a). However, when the effect of the regime variable on a causal variable C_i can be described by a *binary* variable, *i.e.*, $I_i \in \{0, 1\}$, the two representations become distinguishable. In Figure 3, we visualize the two representations by showing the means of the

different variables under interactions, which we detail in Appendix B.6 and provide intuition here. For the original representation C_1, C_2 , each variable’s mean takes on only two different values for any R^t . For example, for regime variables where $I_1 = 0$, the variable C_1 takes a mean that is in the center of the coordinate system. Similarly, when $I_1 = 1$, the variable C_1 will take a mean that is represented as a pink (for $I_1 = 1, I_2 = 0$) or yellow tick (for $I_1 = 1, I_2 = 1$). In contrast, for the rotated variables, both \hat{C}_1 and \hat{C}_2 have three different means depending on the interactions, making it impossible to model them with individual binary variables. Intuitively, the only alternative representations to C_1, C_2 which can be described by binary variables are permutations and/or element-wise transformations, effectively identifying the causal variables according to our identifiability class.

3.3 IDENTIFIABILITY RESULT

When extending this intuition to more than two variables, we find that systems may become unidentifiable when the two distributions of each causal variable, *i.e.*, interacted and not interacted, always differ in the same manner. Formally, we denote the log-likelihood difference between the two distributions of a causal variable C_i^t as $\Delta(C_i^t | C^{t-1}) := \log p(C_i^t | C^{t-1}, I_i^t = 1) - \log p(C_i^t | C^{t-1}, I_i^t = 0)$. If this difference or its derivative w.r.t. C_i^t is constant for all values of C_i^t , the effect of the interactions could be potentially modeled in fewer than K dimensions, giving rise to models that do not identify the causal model \mathcal{M} .

To prevent this, we consider two possible setups for ensuring sufficient variability of $\Delta(C_i^t | C^{t-1})$: *dynamics variability*, and *time variability*. We present our identifiability result below and provide the proofs in Appendix B.

Theorem 3.3. An estimated model $\widehat{\mathcal{M}} = \langle \hat{g}, \hat{f}, \hat{\omega}, \hat{\mathcal{C}} \rangle$ identifies the true causal model $\mathcal{M} = \langle g, f, \omega, \mathcal{C} \rangle$ if:

1. (**Observations**) $\widehat{\mathcal{M}}$ and \mathcal{M} model the same likelihood:

$$p_{\widehat{\mathcal{M}}}(X^t | X^{t-1}, R^t) = p_{\mathcal{M}}(X^t | X^{t-1}, R^t);$$

2. (**Distinct Interaction Patterns**) Each variable C_i in \mathcal{M} has a distinct interaction pattern (Definition 3.2);

and one of the following two conditions holds for \mathcal{M} :

- A. (**Dynamics Variability**) Each variable’s log-likelihood difference is twice differentiable and not always zero:

$$\forall C_i^t, \exists C^{t-1}: \frac{\partial^2 \Delta(C_i^t | C^{t-1})}{\partial (C_i^t)^2} \neq 0;$$

- B. (**Time Variability**) For any $C^t \in \mathcal{C}$, there exist $K+1$ different values of C^{t-1} denoted with $c^1, \dots, c^{K+1} \in \mathcal{C}$, for which the vectors $v_1, \dots, v_K \in \mathbb{R}^{K+1}$ with

$$v_i = \left[\frac{\partial \Delta(C_i^t | C^{t-1}=c^1)}{\partial C_i^t} \quad \dots \quad \frac{\partial \Delta(C_i^t | C^{t-1}=c^{K+1})}{\partial C_i^t} \right]^T$$

are linearly independent.

Intuitively, Theorem 3.3 states that we can identify a causal model \mathcal{M} by maximum likelihood optimization, if we have distinct interaction patterns (Definition 3.2) and $\Delta(C_i^t | C^{t-1})$ varies sufficiently, either in dynamics or in time. *Dynamics variability* can be achieved by the difference $\Delta(C_i^t | C^{t-1})$ being non-linear for all causal variables. This assumption is common in previous ICA-based works (Hyvärinen et al., 2019; Yao et al., 2022a,b) and, for instance, allows for Gaussian distributions with variable standard deviations. While allowing for a variety of distributions, it excludes additive Gaussian noise models. We can include this challenging setup by considering the *time variability* assumption, which states that the effect of the interaction depends on the previous time step, and must do so differently for each variable. As an example, consider a dynamical system with several moving objects, where an interaction is a collision with a robotic arm. The time variability condition is commonly fulfilled by the fact that the trajectory of each object depends on its own velocity and position.

In comparison to previous work, our identifiability results cover a larger class of causal models by exploiting the binary nature of the interaction variables. We provide a detailed comparison in Appendix B.5. In short, closest to our setup, Khemakhem et al. (2020a) and Yao et al. (2022b) require a stronger form of both our dynamics and time variability assumptions, excluding common models like additive Gaussian noise models. Lachapelle et al. (2022b) requires that no two causal variables share the same parents, limiting the allowed temporal graph structures. Meanwhile, our identifiability results allow for arbitrary temporal causal graphs. Further, the two conditions of Theorem 3.3 complement each other well by covering different underlying distributions for the same general setup. Thus, in the next section, we can develop one joint learning algorithm for identifying the causal variables based on both conditions in Theorem 3.3.

4 BISCUIT

Using the results of Section 3, we propose BISCUIT (**B**inary **I**nteractions for **C**ausal **I**dentifiability), a neural-network based approach to identify causal variables and their interaction variables. In short, BISCUIT is a variational autoencoder (VAE) (Kingma et al., 2014), which aims at modeling each of the causal variables C_1, \dots, C_K in a separate latent dimension by enforcing the latent structure of Figure 2. We first give an overview of BISCUIT and then detail the design choices for the model prior.

4.1 OVERVIEW

BISCUIT consists of three main elements: the encoder q_ϕ , the decoder p_θ , and the prior p_ω . The decoder and encoder implement the observation function g and its inverse g^{-1} (Definition 3.1), respectively, and act as a map between ob-

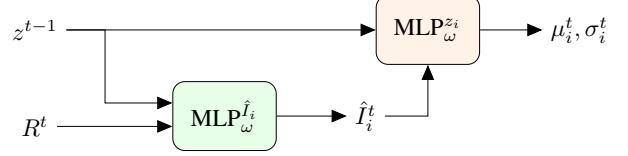


Figure 4: The prior structure of BISCUIT. Based on the previous latents z^{t-1} and the observed regime variable R^t , the $\text{MLP}_{\omega}^{\hat{I}_i}$ predicts the interaction variable \hat{I}_i^t . Then, $\text{MLP}_{\omega}^{z_i}$ outputs the distribution for the next time step $p(z_i^t | z^{t-1}, \hat{I}_i^t)$, which can be parameterized by, e.g., a mean μ_i^t and std σ_i^t .

servations x^t and a lower-dimensional latent space $z^t \in \mathbb{R}^M$, in which we learn the causal variables C_1^t, \dots, C_K^t . The goal of the model is to learn each causal variable C_i^t in a different latent dimension, e.g., z_j^t , effectively separating and hence identifying the causal variables according to Theorem 3.1. Thus, we need the latent space to have at least K dimensions. In practice, since the number of causal variables is not known a priori, we commonly overestimate the latent dimensionality, i.e., $M \gg K$. Still, we expect the model to only use K dimensions actively, with the redundant dimensions not containing any information after training.

On this latent space, the prior p_ω learns a distribution that follows the structure in Definition 3.1, modeling the dynamics in the latent space. As an objective, we maximize the data likelihood of observation triplets $\{X^t, X^{t-1}, R^t\}$ from the true causal model, as stated in Theorem 3.3. The loss function for BISCUIT is:

$$\mathcal{L}^t = -\mathbb{E}_{q_\phi(z^t|x^t)} [\log p_\theta(x^t|z^t)] + \mathbb{E}_{q_\phi(z^{t-1}|x^{t-1})} [\text{KL}(q_\phi(z^t|x^t) || p_\omega(z^t|z^{t-1}, R^t))] \quad (2)$$

with learnable parameter sets ϕ (encoder), θ (decoder), and ω (prior), and KL being the Kullback-Leibler divergence.

For visually complex datasets, the VAE commonly has to perform a trade-off between reconstruction quality and prior modeling, which may cause poorer identification of the causal variables. To circumvent that, we follow Lippe et al. (2022b) by separating the reconstruction and prior modeling stage by training an autoencoder and a normalizing flow (Rezende et al., 2015) in separate stages. In this setup, an autoencoder is first trained to map the observations x^t into a lower-dimensional space. Afterward, we learn a normalizing flow on the autoencoder’s representations to transform them into the desired causal representation, using the same prior structure as in the VAE. In experiments, we refer to this approach as BISCUIT-NF, and the previously described VAE-based approach as BISCUIT-VAE.

4.2 MODEL PRIOR

Our prior follows the distribution structure of Definition 3.1, which has two elements per latent variable: a function to

model the binary interaction variable, and a conditional distribution. We integrate this into BISCUIT’s prior by learning both via multi-layer perceptrons (MLPs):

$$p_{\omega}(z^t | z^{t-1}, R^t) = \prod_{i=1}^M p_{\omega,i}(z_i^t | z^{t-1}, \text{MLP}_{\omega}^{\hat{I}_i}(R^t, z^{t-1})). \quad (3)$$

Here, $\text{MLP}_{\omega}^{\hat{I}_i}$ is an MLP that maps the regime variable R^t and the latents of the previous time step z^{t-1} to a binary output \hat{I}_i^t , as shown in Figure 4. This MLP aims to learn the interaction variable for the latent variable z_i , simply by optimizing Equation (2). The variable \hat{I}_i^t is then used as input for predicting the distribution over z_i^t . For simplicity, we model $p_{\omega,i}$ as a Gaussian distribution, which is parameterized by one MLP per variable, $\text{MLP}_{\omega}^{z_i}$, predicting the mean and standard deviation. To allow for more complex distributions, $p_{\omega,i}$ can alternatively be modeled by a conditional normalizing flow (Winkler et al., 2019).

In early experiments, we found that enforcing \hat{I}_i^t to be a binary variable and backpropagating through it with the straight-through estimator (Bengio et al., 2013) leads to sub-optimal performances. Instead, we model \hat{I}_i^t as a continuous variable during training by using a temperature-scaled tanh as the output activation function of $\text{MLP}_{\omega}^{\hat{I}_i}$. By gradually decreasing the temperature, we bring the activation function closer to a discrete step function towards the end of training.

5 RELATED WORK

Causal Discovery from Unknown Targets Learning (equivalence classes of) causal graphs from observational and interventional data, even with unknown intervention targets, is a common setting in causal discovery (Brouillard et al., 2020; Jaber et al., 2020; Mooij et al., 2020; Squires et al., 2020). In recent work, this is even extended to the case in which we have unknown mixtures of interventional data (Faria et al., 2022; Kumar et al., 2021; Mian et al., 2023), which for example can happen if the regime variable is not observed. In our paper, we assume that we observe the regime variable and then reconstruct the latent interaction variables, which resemble the observed context variables by Mooij et al. (2020). Moreover, our work is on a different task, causal representation learning, in which we try to learn the causal variables from high-dimensional data.

Causal Representation Learning A common basis for causal representation learning is Independent Component Analysis (ICA) (Comon, 1994; Hyvärinen et al., 2001), which aims to identify independent latent variables from observations. Due to the non-identifiability for the general case of non-linear observation functions (Hyvärinen et al., 1999), additional auxiliary variables are often considered in this setting (Hyvärinen et al., 2016; Hyvärinen et al., 2019). Ideas from ICA have been integrated into neural networks (Khemakhem et al., 2020a,b; Reizinger et al., 2022) and

applied to causality (Gresele et al., 2021; Monti et al., 2019; Shimizu et al., 2006) for identifying causal variables.

Recently, several works in causal representation learning have exploited distribution shifts or interventions to identify causal variables. Using counterfactual observations, Ahuja et al. (2022); Brehmer et al. (2022); Locatello et al. (2020) learn causal variables from pairs of images, between which only a subset of variables has changed via interventions with unknown targets. For temporal processes, Lachapelle et al. (2022a,b) can model interventions of unknown target via *actions*, which is equivalent to the regime variable in our setting, but require that each causal variable has a strictly unique parent set. On the other hand, Yao et al. (2022a,b) consider observations from m different regimes u_1, \dots, u_m , where, in our setting, the regime indicator u is a discrete version of R^t . However, they require at least $2K + 1$ different regimes compared to $\lfloor \log_2 K \rfloor + 2$ settings for ours, and have stronger conditions on the distribution changes over regimes (*e.g.*, no additive Gaussian noise models). In temporal settings where the intervention targets are known, CITRIS (Lippe et al., 2022b, 2023) identifies scalar and multidimensional causal variables from high-dimensional images. Nonetheless, observing the intervention targets requires additional supervision, which may not always be available. To the best of our knowledge, we are the first to use unknown binary interactions to identify the causal variables from high-dimensional observations.

6 EXPERIMENTS

To illustrate the effectiveness of BISCUIT, we evaluate it on a synthetic toy benchmark and two environments generated by 3D robotic simulators. We publish our code at <https://github.com/phlippe/BISCUIT>, and detail the data generation and hyperparameters in Appendix C.

6.1 SYNTHETIC TOY BENCHMARK

To evaluate BISCUIT on various graph structures, we extend the Voronoi benchmark (Lippe et al., 2022b) by replacing observed intervention targets with unobserved binary interactions. In this dataset, each causal variable follows an additive Gaussian noise model, where the mean is modeled by a randomly initialized MLP. To determine the parent set, we randomly sample the causal graph with an edge likelihood of 0.4. Instead of observing the causal variables directly, they are first entangled by applying a two-layer randomly initialized normalizing flow before visualizing the outputs as colors in a Voronoi diagram of size 32×32 (see Figure 5a). We extend the original benchmark by including a robotic arm that moves over the Voronoi diagram and interacts by touching individual color regions/tiles. Each tile corresponds to one causal variable, allowing for both single- and multi-target interactions. The models need to deduce

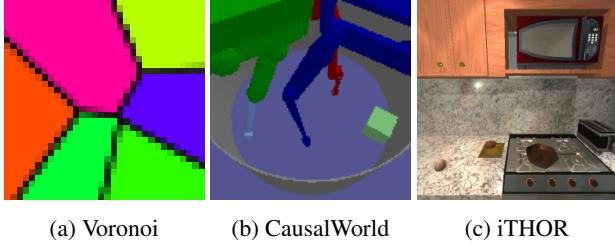


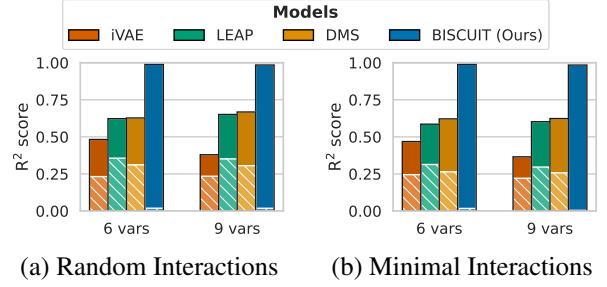
Figure 5: Example figures of our three environments with increasing complexity: Voronoi (Lippe et al., 2023), CausalWorld (Ahmed et al., 2020), and iTHOR (Kolve et al., 2017).

these interactions from a regime variable $R^t \in [0, 1]^2$ which is the 2D location of the robotic arm on the image. When the robotic arm interacts with a variable, its mean is set to zero, which resembles a stochastic perfect intervention.

Evaluation We generate five Voronoi systems with six causal variables, and five systems with nine variables. We compare BISCUIT to iVAE (Khemakhem et al., 2020a), LEAP (Yao et al., 2022b), and Disentanglement via Mechanism Sparsity (DMS) (Lachapelle et al., 2022b), since all use a regime variable. We do not compare with CITRIS (Lippe et al., 2022b, 2023), because it requires known intervention targets. We follow Lippe et al. (2023) in evaluating the models on a held-out test set where all causal variables are independently sampled. We calculate the coefficient of determination (Wright, 1921), also called the R^2 score, between each causal variable C_i and each learned latent variable z_j , denoted by R_{ij}^2 . If a model identifies the causal variables according to Definition 3.1, then for each causal variable C_i , there exists one latent variable z_j for which $R_{ij}^2 = 1$, while it is zero for all others. Since the alignment of the learned latent variables to causal variables is not known, we report R^2 scores for the permutation π that maximizes the diagonal of the R^2 matrix, *i.e.*, $R^2\text{-diag} = 1/K \sum_{i=1}^K R_{i,\pi(i)}^2$ (where 1 is optimal). To account for spurious modeled correlation, we also report the maximum correlation besides this alignment: $R^2\text{-sep} = 1/K \sum_{i=1}^K \max_{j \neq \pi(i)} R_{ij}^2$ (optimal 0).

Results The results in Figure 6a show that BISCUIT identifies the causal variables with high accuracy for both graphs with six and nine variables. In comparison, all baselines struggle to identify the causal variables, often falling back to modeling the colors as latent variables instead. While the assumptions of iVAE and LEAP do not hold for additive Gaussian noise models, the assumptions of DMS, including the graph sparsity, mostly hold. Still, BISCUIT is the only method to consistently identify the true variables, illustrating that its stable optimization and robustness.

Minimal Number of Regimes To verify that BISCUIT only requires $\lfloor \log_2 K \rfloor + 2$ different regimes (Theorem 3.3), we repeat the previous experiments with reducing the interaction maps to a minimum. This results in four sets of interactions for six variables, and five for nine variables. Fig-



(a) Random Interactions (b) Minimal Interactions

Figure 6: Results on the Voronoi benchmark averaged over 10 seeds. Solid bars show the mean R^2 -diag score (higher is better), and striped bars the R^2 -sep scores (lower is better, non-visible bars indicate close-to zero values). BISCUIT accurately identifies the causal variables across settings.

ure 6b shows that BISCUIT still correctly identifies causal variables in this setting, supporting our theoretical results.

Learned Intervention Targets After training, we can use the interaction variables $\hat{I}_1, \dots, \hat{I}_M$ learned by BISCUIT to identify the regions in which the robotic arm interacts with a causal variable. Based on our theoretical results, we expect that some of the learned variables are identical to the true interaction variables I_1, \dots, I_K up to permutations and sign-flips. In all settings, we find that the learned binary variables match the true interaction variables with an average F1 score of 98% for the same permutation of variables as in the R^2 evaluation. This shows that BISCUIT identified the true interaction variables. Thus, in practice, one could use a few samples with labeled interaction variables to identify the learned permutation of the model.

6.2 CAUSALWORLD

CausalWorld (Ahmed et al., 2020) is a robotic manipulation environment with a tri-finger robot, which can interact with objects in an enclosed space by touch (see Figure 5b). The environment also allows for interventions on various environment parameters, including the colors or friction parameters of individual elements. We experiment on this environment by recording the robot’s interactions with a cube. Besides the cube position, rotation and velocity, the causal variables are the colors of the three fingertips, as well as the floor, stage and cube friction, which we visualize by the colors of the respective objects. All colors and friction parameters follow an additive Gaussian noise model. When a robot finger touches the cube, we perform a stochastic perfect intervention on its color. Similarly, an interaction with the friction parameters correspond to touching these objects with all three fingers. The regime variable R^t is modeled by the angles of the three motors per robot finger from the current and previous time step, providing velocity information.

This environment provides two new challenges. Firstly, not

Table 1: R^2 scores (diag \uparrow / sep \downarrow) for the identification of the causal variables on CausalWorld and iTHOR.

Models	CausalWorld	iTHOR
iVAE (Khemakhem et al., 2020a)	0.28 / 0.00	0.48 / 0.35
LEAP (Yao et al., 2022b)	0.30 / 0.00	0.63 / 0.45
DMS (Lachapelle et al., 2022b)	0.32 / 0.00	0.61 / 0.40
BISCUIT-NF (Ours)	0.97 / 0.01	0.96 / 0.15

all interactions are necessarily binary. In particular, the collisions between the robot and the cube have different effects depending on the velocity and direction of the fingers of the robot, which are not part of the state of the causal variables at the previous time step. Additionally, the robotic system is present in the observation/image, while our theoretical results assume that R^t is not a direct cause of X^t . We adapt BISCUIT-NF and the baselines to this case by adding R^t as additional information to the decoder, effectively removing the need to model R^t in the latent space.

On this task, BISCUIT identifies the causal variables well, as seen in Table 1. Because the cube position, velocity and rotation share the same interactions, in the evaluation we consider them as a multidimensional variable. Although the true model cannot be fully described by binary interaction variables, BISCUIT still models the binary information of whether a collision happens or not for the cube, since it is the most important part of the dynamics. We verify this in Appendix C.2.3 by measuring the F1 score between the predicted interaction variables and ground truth interactions/collisions. BISCUIT achieves an F1 score of 50% for all cube-arm interactions, which indicates a high similarity between the learned interaction and the ground truth collisions considering that collisions only happen in approximately 5% of the frames. The mismatches are mostly due to the learned interactions being more conservative, *i.e.*, being 1 already a frame too early sometimes. Meanwhile, none of the baselines are able to reconstruct the image sufficiently, missing the robotic arms and the cube (see Appendix C.2.3). While this might improve with significant tuning effort, BISCUIT-NF is not sensitive to the difficulty of the reconstruction due to its separate autoencoder training stage.

6.3 ITHOR - EMBODIED AI

To illustrate the potential of causal representation learning in embodied AI, we apply BISCUIT to the iTHOR environment (Kolve et al., 2017). In this environment, an embodied AI agent can perform actions on various objects in an 3D indoor scene such as a kitchen. These agent-object interactions can often be described by a binary variable, *e.g.*, pickup/put down an object, open/close a door, turn on/off an object, etc., which makes it an ideal setup for BISCUIT.

Our goal in this environment is to identify the causal vari-

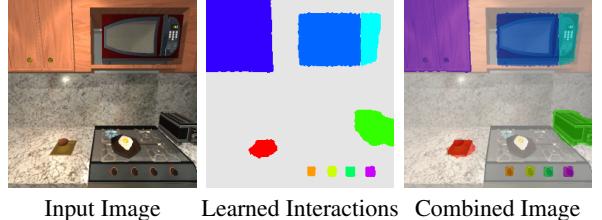


Figure 7: Visualizing the learned interaction variables of BISCUIT for an example input image (left). We show the locations, *i.e.*, values of $R^t \in [0, 1]^2$, for which each interaction variable is greater than zero/active as different colors. For readability, only nine interaction variables are shown. The right image is an overlay of both. BISCUIT accurately learns the interactions and adapts them to the input image.

ables, *i.e.*, the objects and their states, from sequences of interactions. We perform this task on the kitchen environment shown in Figure 5c. This environment contains two movable objects, *i.e.*, a plate and an egg, and seven static objects, *e.g.*, a microwave and a stove. Overall, we have 18 causal variables, which include both continuous, *e.g.*, the location of the plate, and binary variables, *e.g.*, whether the microwave is on or off. Causal variables influence each other by state changes, *e.g.*, the egg gets cooked when it is in the pan and the stove is turned on. Further, the set of possible actions that can be performed depends on the previous time step, *e.g.*, only one object can be picked up at a time. For training, we generate a dataset where we randomly pick a valid action at each time step. We model the regime variable R^t as a two-dimensional pixel coordinate, which is the position of a pixel showing the interacted object in the image ($R^t \in [0, 1]^2$). This simulates iTHOR’s web demo (Kolve et al., 2017), where a user interacts with objects by clicking on them.

We train BISCUIT-NF and our baselines on this dataset, and compare the latent representation to the ground truth causal variables in terms of the R^2 score in Table 1. Although the baselines reconstruct the image mostly well, the causal variables are highly entangled in their representations. In contrast, BISCUIT identifies and separates most of the causal variables optimally, except for the two movable objects (egg/plate). This is likely due to the high inherent correlation of the two objects, since their positions cannot overlap and only one of them can be picked up at a time.

Besides evaluating the causal representation, we also visualize the learned interaction variables of BISCUIT in Figure 7. Here, each color represents the region in which BISCUIT identified an interaction with a different causal variable. Figure 7 shows that BISCUIT has identified the correct interaction region for each object. Moreover, it allows for context-dependent interactions, as the location of the plate influences the region of its corresponding interaction variable.

Finally, we can use the learned causal representation to per-



Figure 8: Performing interventions in the latent space. First, the two inputs images are encoded into latent space. Then, we replace the latents of the front-left stove and microwave in the first image by the corresponding latents of the second image. This corresponds to effectively turning on the stove and turning off the microwave in the first image. All remaining latents are kept fixed, aiming the other causal variables to remain unchanged from their state in the first image. Decoding these new latents in the right image shows that BISCUIT can model the effect of interventions accurately. Furthermore, it is able to create the novel, unseen scenario where the egg is uncooked, but the stove is turned on. This highlights the modularity of BISCUIT’s representations.

form interventions and create novel combinations of causal variables. For this, we encode two images into the learned latent space of BISCUIT, and combine the latent representations of the causal variables to have a novel image decoded. For example, in Figure 8, we replace the latents representing the front-left stove and the microwave state in the first image by the corresponding latents of the second image. This aims to perform an intervention on the front-left stove (turning on) and the microwave state (turning off) while all remaining causal variables such as the egg state should stay unchanged.¹ BISCUIT not only integrates these changes without influencing any of the other causal variables, but generates a completely novel state: even though in the iTHOR environment, the egg is instantaneously cooked when the stove turns on, BISCUIT correctly combines the state of the egg being raw with the stove burning. This shows the capabilities of BISCUIT to model unseen causal interventions.

7 CONCLUSION

We prove that under mild assumptions, causal variables become identifiable from high-dimensional observations, when their interactions with an external system can be described by unknown binary variables. As a practical algorithm, we propose BISCUIT, which learns the causal variables and their interaction variables. In experiments across three robotic-inspired datasets, BISCUIT outperforms previous methods in identifying the causal variables from images.

¹This setup can also be interpreted as performing a perfect intervention on all causal variables with values picked from image 1 or 2 for individual variables. Causal relations between variables, e.g., the stove and the egg, are actively broken in this setup.

While in experiments, BISCUIT shows strong identification even for complex interactions, the presented theory is currently limited to binary interaction variables. Although the first step may be to generalize the theory to interaction variables with more than two states, extensions to unknown domains or sparse, continuous interaction variables are other interesting future directions. Instead of assuming distinct interaction patterns, future work can extend these results to partial identifiability, similar to Lachapelle et al. (2022a); Lippe et al. (2022b). Finally, our results open up the opportunity for empirical studies showing the benefits of causal representations for complex real-world tasks like embodied AI.

Author Contributions

P. Lippe conceived the idea, developed the theoretical results, implemented the models and experiments, and wrote the paper. S. Magliacane, S. Löwe, Y. M. Asano, T. Cohen, E. Gavves advised during the project and helped in writing the paper.

Acknowledgements

We thank SURF for the support in using the National Supercomputer Snellius. This work is financially supported by Qualcomm Technologies Inc., the University of Amsterdam and the allowance Top consortia for Knowledge and Innovation (TKIs) from the Netherlands Ministry of Economic Affairs and Climate Policy.

REFERENCES

- Ahmed, O., Träuble, F., Goyal, A., Neitz, A., Bengio, Y., Schölkopf, B., Wüthrich, M., and Bauer, S. CausalWorld: A robotic manipulation benchmark for causal structure and transfer learning. *arXiv preprint arXiv:2010.04296*, 2020.
- Ahuja, K., Hartford, J., and Bengio, Y. Weakly Supervised Representation Learning with Sparse Perturbations. In *Advances in Neural Information Processing Systems 35, NeurIPS*, 2022.
- Assaad, C. K., Devijver, E., and Gaussier, E. Survey and evaluation of causal discovery methods for time series. *J. Artif. Int. Res.*, 73, may 2022. ISSN 1076-9757.
- Bengio, Y., Léonard, N., and Courville, A. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.
- Brehmer, J., de Haan, P., Lippe, P., and Cohen, T. Weakly supervised causal representation learning. In *Advances in Neural Information Processing Systems 35, NeurIPS*, 2022.

- Brouillard, P., Lachapelle, S., Lacoste, A., Lacoste-Julien, S., and Drouin, A. Differentiable Causal Discovery from Interventional Data. In *Advances in Neural Information Processing Systems 33, NeurIPS*, 2020.
- Comon, P. Independent component analysis, A new concept? *Signal Processing*, 36(3), April 1994.
- Dean, T. and Kanazawa, K. A model for reasoning about persistence and causation. *Computational Intelligence*, 5(2), 1989.
- Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using Real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.
- Falcon, W. and The PyTorch Lightning team. PyTorch Lightning, 2019.
- Faria, G. R. A., Martins, A., and Figueiredo, M. A. T. Differentiable Causal Discovery Under Latent Interventions. In *Proceedings of the First Conference on Causal Learning and Reasoning*, volume 177 of *Proceedings of Machine Learning Research*. PMLR, 11–13 Apr 2022.
- Gresele, L., von Kügelgen, J., Stimper, V., Schölkopf, B., and Besserve, M. Independent mechanism analysis, a new concept? In *Advances in Neural Information Processing Systems*, 2021.
- Hafner, D., Lillicrap, T. P., Norouzi, M., and Ba, J. Mastering Atari with Discrete World Models. In *International Conference on Learning Representations*, 2021.
- He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.
- Hyvärinen, A. and Morioka, H. Unsupervised Feature Extraction by Time-Contrastive Learning and Nonlinear ICA. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*. Curran Associates Inc., 2016.
- Hyvärinen, A. and Pajunen, P. Nonlinear Independent Component Analysis: Existence and Uniqueness Results. *Neural Netw.*, 12(3), apr 1999. ISSN 0893-6080.
- Hyvärinen, A., Karhunen, J., and Oja, E. *Independent Component Analysis*. John Wiley & Sons, June 2001.
- Hyvärinen, A., Sasaki, H., and Turner, R. Nonlinear ICA Using Auxiliary Variables and Generalized Contrastive Learning. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*. PMLR, 2019.
- Jaber, A., Kocaoglu, M., Shanmugam, K., and Bareinboim, E. Causal Discovery from Soft Interventions with Unknown Targets: Characterization and Learning. In *Advances in Neural Information Processing Systems 33, NeurIPS*, 2020.
- Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. Variational Autoencoders and Nonlinear ICA: A Unifying Framework. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*. PMLR, 2020a.
- Khemakhem, I., Monti, R., Kingma, D., and Hyvarinen, A. ICE-BeeM: Identifiable Conditional Energy-Based Deep Models Based on Nonlinear ICA. In *Advances in Neural Information Processing Systems 33, NeurIPS*, 2020b.
- Kingma, D. P. and Ba, J. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.
- Kingma, D. P. and Dhariwal, P. Glow: Generative Flow with Invertible 1x1 Convolutions. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.
- Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.
- Klindt, D., Schott, L., Sharma, Y., Ustyuzhaninov, I., Brendel, W., Bethge, M., and Paiton, D. Towards Nonlinear Disentanglement in Natural Data with Temporal Sparse Coding. In *International Conference on Learning Representations (ICLR)*, 2021.
- Kolve, E., Mottaghi, R., Han, W., VanderBilt, E., Weihs, L., Herrasti, A., Gordon, D., Zhu, Y., Gupta, A., and Farhadi, A. AI2-THOR: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*, 2017. Web demo <https://ai2thor.allenai.org/demo/>.
- Kumar, A. and Sinha, G. Disentangling mixtures of unknown causal interventions. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of *Proceedings of Machine Learning Research*. PMLR, 27–30 Jul 2021.
- Lachapelle, S. and Lacoste-Julien, S. Partial Disentanglement via Mechanism Sparsity. In *UAI 2022 Workshop on Causal Representation Learning*, 2022a.
- Lachapelle, S., Rodriguez, P., Le, R., Sharma, Y., Everett, K. E., Lacoste, A., and Lacoste-Julien, S. Disentanglement via Mechanism Sparsity Regularization: A New Principle for Nonlinear ICA. In *First Conference on Causal Learning and Reasoning*, 2022b.

- Lesort, T., Díaz-Rodríguez, N., Goudou, J.-F., and Filliat, D. State representation learning for control: An overview. *Neural Networks*, 108, 2018.
- Lippe, P., Cohen, T., and Gavves, E. Efficient Neural Causal Discovery without Acyclicity Constraints. In *International Conference on Learning Representations*, 2022a.
- Lippe, P., Magliacane, S., Löwe, S., Asano, Y. M., Cohen, T., and Gavves, E. CITRIS: Causal Identifiability from Temporal Intervened Sequences. In *Proceedings of the 39th International Conference on Machine Learning, ICML*, 2022b.
- Lippe, P., Magliacane, S., Löwe, S., Asano, Y. M., Cohen, T., and Gavves, E. Intervention Design for Causal Representation Learning. In *UAI 2022 Workshop on Causal Representation Learning*, 2022c.
- Lippe, P., Magliacane, S., Löwe, S., Asano, Y. M., Cohen, T., and Gavves, E. Causal representation learning for instantaneous and temporal effects in interactive systems. In *The Eleventh International Conference on Learning Representations*, 2023.
- Locatello, F., Poole, B., Rätsch, G., Schölkopf, B., Bachem, O., and Tschannen, M. Weakly-Supervised Disentanglement Without Compromises. In *Proceedings of the 37th International Conference on Machine Learning, ICML*, 2020.
- Mian, O., Kamp, M., and Vreeken, J. Information-theoretic causal discovery and intervention detection over multiple environments. In *Proceedings of the AAAI Conference on Artificial Intelligence, AAAI-23*, 2023.
- Monti, R. P., Zhang, K., and Hyvärinen, A. Causal Discovery with General Non-Linear Relationships using Non-Linear ICA. In *Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI*, 2019.
- Mooij, J. M., Magliacane, S., and Claassen, T. Joint Causal Inference from Multiple Contexts. *Journal of Machine Learning Research*, 21(99), 2020.
- Murphy, K. Dynamic Bayesian Networks: Representation, Inference and Learning. *UC Berkeley, Computer Science Division*, 2002.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 2019.
- Peters, J., Janzing, D., and Schölkopf, B. Causal inference on time series using restricted structural equation models. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. (eds.), *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.
- Ramachandran, P., Zoph, B., and Le, Q. V. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.
- Reizinger, P., Gresele, L., Brady, J., von Kügelgen, J., Zietlow, D., Schölkopf, B., Martius, G., Brendel, W., and Besserve, M. Embrace the Gap: VAEs Perform Independent Mechanism Analysis. In *Advances in Neural Information Processing Systems 35, NeurIPS*, 2022.
- Rezende, D. J. and Mohamed, S. Variational Inference with Normalizing Flows. In *Proceedings of the 32nd International Conference on Machine Learning, ICML*, 2015.
- Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., and Bengio, Y. Toward causal representation learning. *Proceedings of the IEEE*, 109(5), 2021.
- Shimizu, S., Hoyer, P. O., Hyvärinen, A., and Kerminen, A. A Linear Non-Gaussian Acyclic Model for Causal Discovery. *J. Mach. Learn. Res.*, 7, dec 2006.
- Squires, C., Wang, Y., and Uhler, C. Permutation-based causal structure learning with unknown intervention targets. In *Conference on Uncertainty in Artificial Intelligence*. PMLR, 2020.
- Träuble, F., Dittadi, A., Wuthrich, M., Widmaier, F., Gehler, P. V., Winther, O., Locatello, F., Bachem, O., Schölkopf, B., and Bauer, S. The Role of Pretrained Representations for the OOD Generalization of RL Agents. In *International Conference on Learning Representations*, 2022.
- Winkler, C., Worrall, D., Hoogeboom, E., and Welling, M. Learning likelihoods with conditional normalizing flows. *arXiv preprint arXiv:1912.00042*, 2019.
- Wright, S. Correlation and causation. *Journal of agricultural research*, 20(7), 1921.
- Wu, Y. and He, K. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.
- Yao, W., Chen, G., and Zhang, K. Temporally Disentangled Representation Learning. In *Advances in Neural Information Processing Systems 35, NeurIPS*, 2022a.
- Yao, W., Sun, Y., Ho, A., Sun, C., and Zhang, K. Learning Temporally Causal Latent Processes from General Temporal Data. In *International Conference on Learning Representations*, 2022b.

Zheng, X., Aragam, B., Ravikumar, P., and Xing, E. P. DAGs with NO TEARS: Continuous Optimization for Structure Learning. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, 2018.

BISCUIT: Causal Representation Learning from Binary Interactions

Appendix

TABLE OF CONTENTS

A Reproducibility Statement	14
B Proofs	14
B.1 Glossary	14
B.2 Proof Steps	14
B.3 Extension to Longer Temporal Dependencies	25
B.4 Identifying the Temporal Causal Graph	26
B.5 Relation to Previous Identifiability Results	26
B.6 Detailed Example for Additive Gaussian Noise	27
C Experimental Details	29
C.1 Voronoi	29
C.2 CausalWorld	32
C.3 iTHOR	36

A REPRODUCIBILITY STATEMENT

For reproducibility, we publish the code of BISCUIT and the generations of the datasets (Voronoi, CausalWorld, iTHOR) as well as the datasets itself at <https://github.com/phlippe/BISCUIT>. All models were implemented using PyTorch (Paszke et al., 2019) and PyTorch Lightning (Falcon et al., 2019). The hyperparameters and dataset details are described in Section 6 and Appendix C. All experiments have been repeated for at least three seeds. We provide an overview of the standard deviation, as well as additional insights to the results in Appendix C.

In terms of computational resources, the experiments of the Voronoi dataset were performed on a single NVIDIA A5000 GPU, with a training time of below 1 hour per model. The experiments of the CausalWorld and iTHOR dataset were performed on an NVIDIA A100 GPU (autoencoder training: 1-day training time; variational autoencoders: 1 to 2-day training time) and an NVIDIA A5000 GPU (normalizing flow training, 1-hour training time).

B PROOFS

In this section, we prove the main theoretical results in this paper, namely Theorem 3.3. We start with a glossary of the used notation in Appendix B.1, which is the same as in the main paper. All assumptions for the proof are described in Section 2 and Section 3 of the main paper. Appendix B.2 contains the proofs for Theorem 3.3. Lastly, we provide further discussion on extensions of the proof, *e.g.*, to longer temporal dependencies (Appendix B.3), discovering the causal graph (Appendix B.4), and comparing its results to previous work (Appendix B.5).

B.1 GLOSSARY

Table 2 provides an overview of the main notation used in the paper and the following proof. Additional notation for individual proof steps is introduced in the respective sections.

Table 2: Glossary of used functions and variables.

Function/Variable	Description
\mathcal{M}	The true causal model $\mathcal{M} = \langle g, f, \omega, \mathcal{C} \rangle$
K	Number of causal variables
C_1, \dots, C_K	The causal variables of \mathcal{M}
C_1^t, \dots, C_K^t	Instantiations of causal variables of \mathcal{M} at time t
\mathcal{C}	Domain of the causal variables, <i>i.e.</i> , $C^t \in \mathcal{C}$
R	The regime variable
I_1, \dots, I_K	Binary interaction variables
f_1, \dots, f_K	Functions determining the interaction variables, $f_i(R^t, C^{t-1}) = I_i^t$
$p(C^t C^{t-1}, I^t)$	Conditional distribution of causal variables
ω	Parameters of the conditional distribution
$\Delta(C_i^t C^{t-1})$	Interaction effect for the causal variable C_i : $\Delta(C_i^t C^{t-1}) = \log \frac{p(C_i^t C^{t-1}, I_i^t = 1)}{p(C_i^t C^{t-1}, I_i^t = 0)}$
X^t	The observation, <i>e.g.</i> , an image
g	The observation function $g(C^t) = X^t$ of \mathcal{M}
$\widehat{\mathcal{M}}$	Causal model $\widehat{\mathcal{M}} = \langle \hat{g}, \hat{f}, \hat{\omega}, \hat{\mathcal{C}} \rangle$ with the same data distribution over observations X as \mathcal{M}
$\hat{C}_1, \dots, \hat{C}_K$	Causal variables modeled by a model $\widehat{\mathcal{M}}$ with domain $\hat{\mathcal{C}}$
$\hat{I}_1, \dots, \hat{I}_K$	Binary interaction variables modeled by a model $\widehat{\mathcal{M}}$
\hat{g}	(Estimated) observation function of the model $\widehat{\mathcal{M}}$, $g(\hat{C}^t) = X^t$

B.2 PROOF STEPS

The proof consists of four main steps:

1. We show that for any \hat{g} in $\widehat{\mathcal{M}}$, there must exist an invertible transformation between the latent space of $\widehat{\mathcal{M}}$ and the true causal model \mathcal{M} (Appendix B.2.1).

2. We show that the distributions of different interaction variable values must strictly be different, starting with two variables (Appendix B.2.2) and then moving to the general case (Appendix B.2.3).
3. Based on the previous step, we show that \hat{g} must model the same interaction variable patterns for the individual interaction cases, starting with two variables Appendix B.2.4. Appendix B.2.5 discusses it under the assumption of dynamics variability in Theorem 3.3, and Appendix B.2.6 for time variability in Theorem 3.3.
4. Given that both $\widehat{\mathcal{M}}$ and \mathcal{M} model the same interaction variables, the invertible transformation must be equal to a set of component-wise transformations (Appendix B.2.7).

In Appendix B.2.8, we combine these four steps to prove the full theorem. For step 2 and 3, we first start by discussing the proof idea for a system with only two variables, to give a better intuition behind the proof strategy.

B.2.1 Existence of an invertible transformation between learned and true representations

As a first step, we start with discussing the relation between the true observation function, g , and a potentially learned observation function, \hat{g} and show that there exists an invertible transformation between the causal variables that are extracted from an image for each of these two functions. Throughout this proof, we will use C_i to refer to a causal variable from the original, true causal model \mathcal{M} , and we use \hat{C}_i to refer to a latent variable modeled by $\widehat{\mathcal{M}}$, i.e., an alternative representation of the environment. Under this setup, we consider the following statement:

Lemma B.1. *Consider a model $\widehat{\mathcal{M}} = \langle \hat{g}, \hat{f}, \hat{\omega}, \hat{\mathcal{C}} \rangle$ with an injective observation function $g(\hat{C}^t) = X^t$ with $\hat{C}^t \in \hat{\mathcal{C}}$ and a latent distribution $p_\omega(\hat{C}^t | \hat{C}^{t-1}, R^t)$, parameterized by ω , which models the same data likelihood as the true causal model \mathcal{M} : $p_{\mathcal{M}}(X^t | X^{t-1}, R^t) = p_{\widehat{\mathcal{M}}}(X^t | X^{t-1}, R^t)$. Then, there must exist an invertible transformation T such that for all C^t, \hat{C}^t :*

$$C^t = T(\hat{C}^t). \quad (4)$$

Proof. Since g and \hat{g} are injective functions with $\hat{g}(\hat{C}^t) = X^t$ and $g(C^t) = X^t$, we have that $X^t = (g \circ g^{-1})(X^t) = (\hat{g} \circ \hat{g}^{-1})(X^t) = \text{id}_X(X^t)$ with id being the identity function, when restricting the injective functions to their ranges to turn them to invertible functions with $C^t = (g^{-1} \circ g)(C^t) = \text{id}_C(C^t)$ and $\hat{C}^t = (\hat{g}^{-1} \circ \hat{g})(\hat{C}^t) = \text{id}_{\hat{C}}(\hat{C}^t)$. Combining the two results in:

$$C^t = g^{-1}(X^t) = (g^{-1} \circ \hat{g} \circ \hat{g}^{-1})(X^t) = T(\hat{g}^{-1}(X^t)) = T(\hat{C}^t) \quad (5)$$

where $T = g^{-1} \circ \hat{g}$. This function has the inverse $T^{-1} = \hat{g}^{-1} \circ g$ with:

$$T^{-1} \circ T = \hat{g}^{-1} \circ \underbrace{g \circ g^{-1}}_{\text{id}_X} \circ \hat{g} = \hat{g}^{-1} \circ \hat{g} = \text{id}_{\hat{C}} \quad (6)$$

Hence, there exists an invertible function $T = g^{-1} \circ \hat{g}$ between the two spaces of \hat{C}^t and C^t . \square

Since there exists an invertible, differentiable transformation between the two spaces, we can also express the relation between the two spaces via the change-of-variables:

$$p_\omega(C^t | C^{t-1}, R^t) = p_{\hat{\omega}}(\hat{C}^t | \hat{C}^{t-1}, R^t) |\det \mathbf{J}| \quad (7)$$

where \mathbf{J} is the Jacobian with $J_{ij} = \frac{\partial \hat{C}_i^t}{\partial C_j^t}$. Further, since there exists an invertible transformation also between C^{t-1} and \hat{C}^{t-1} , we can align the conditioning set:

$$p_\omega(C^t | C^{t-1}, R^t) = p_{\hat{\omega}}(\hat{C}^t | C^{t-1}, R^t) |\det \mathbf{J}| \quad (8)$$

For readability, we will drop ω and $\hat{\omega}$ from the index of p , since the difference is clear from the context (distribution over C/\hat{C}). The following proof steps will take a closer look at aligning these two spaces.

B.2.2 Any representation requires the same interaction cases - 2 variables

Setup The idea of this proof step is to show that for any two causal variables C_1, C_2 , any representation that models the same data likelihood, e.g., \hat{C}_1, \hat{C}_2 , must have an invertible transformation between the interaction variables I_1^t, I_2^t

(and C_1^{t-1}, C_2^{t-1}) and the learned interaction variables \hat{I}_1^t, \hat{I}_2^t (and $\hat{C}_1^{t-1}, \hat{C}_2^{t-1}$). In other words, disentanglement requires distinguishing between the same scenarios of interactions.

We start with considering the four possible interaction cases that we may encounter:

$$p(C_1^t, C_2^t | C^{t-1}, I_1^t = 0, I_2^t = 0) = p(C_1^t | C^{t-1}, I_1^t = 0) \cdot p(C_2^t | C^{t-1}, I_2^t = 0) \quad (9)$$

$$p(C_1^t, C_2^t | C^{t-1}, I_1^t = 1, I_2^t = 0) = p(C_1^t | C^{t-1}, I_1^t = 1) \cdot p(C_2^t | C^{t-1}, I_2^t = 0) \quad (10)$$

$$p(C_1^t, C_2^t | C^{t-1}, I_1^t = 0, I_2^t = 1) = p(C_1^t | C^{t-1}, I_1^t = 0) \cdot p(C_2^t | C^{t-1}, I_2^t = 1) \quad (11)$$

$$p(C_1^t, C_2^t | C^{t-1}, I_1^t = 1, I_2^t = 1) = p(C_1^t | C^{t-1}, I_1^t = 1) \cdot p(C_2^t | C^{t-1}, I_2^t = 1) \quad (12)$$

Our goal is to show that all these four distributions must be strictly different for any C^{t-1} . These inequalities generalize to any alternative representation that entangles the two variables C_1, C_2 , since the alternative representation must model the same distributions $p(X^t | X^{t-1}, \dots) = p(C^t | \dots) = p(C_1^t, C_2^t | \dots) \cdot \dots$.

Implications of theorem assumptions Before comparing the distributions, we first simplify what the assumptions of Theorem 3.3 imply for the individual variable's distributions. The theorem assumes that $\Delta(C_i^t | C^{t-1}) = \log p(C_i^t | C^{t-1}, I_i^t = 1) / p(C_i^t | C^{t-1}, I_i^t = 0)$ is differentiable and cannot be a constant. Otherwise, the derivatives $\Delta(C_i^t | C^{t-1})$ would have to be constantly zero, which violates both condition (A) and (B) of the theorem. Therefore, we can deduce that:

- For each variable C_i , there must exist at least one value of C_i^t for which $p(C_i^t | C^{t-1}, I_i^t = 1) \neq p(C_i^t | C^{t-1}, I_i^t = 0)$, *i.e.*, $p(C_i^t | C^{t-1}, I_i^t = 1), p(C_i^t | C^{t-1}, I_i^t = 0)$ must strictly be different distributions.
- The distributions $p(C_i^t | C^{t-1}, I_i^t = 1), p(C_i^t | C^{t-1}, I_i^t = 0)$ must share the same support, since otherwise $\Delta(C_i^t | C^{t-1}) = \pm\infty$ for some C_i^t and thus not differentiable.

Single-target vs Joint We start with comparing single-target interactions versus the observational case. Since the interactional distribution is strictly different from the observational, we obtain that $p(C_i^t | C^{t-1}, I_i^t = 0) \neq p(C_i^t | C^{t-1}, I_i^t = 1)$. With this inequality, we can deduce that:

$$p(C_1^t, C_2^t | C^{t-1}, I_1^t = 0, I_2^t = 0) \neq p(C_1^t, C_2^t | C^{t-1}, I_1^t = 1, I_2^t = 0) \quad (13)$$

$$p(C_1^t, C_2^t | C^{t-1}, I_1^t = 0, I_2^t = 0) \neq p(C_1^t, C_2^t | C^{t-1}, I_1^t = 0, I_2^t = 1) \quad (14)$$

This is because these distributions only differ in one sub-distribution (*i.e.*, either C_1 or C_2 intervened versus passively observed), which must be strictly different due to our assumption. A similar reasoning can be used to derive the same inequalities for the joint interaction case:

$$p(C_1^t, C_2^t | C^{t-1}, I_1^t = 1, I_2^t = 1) \neq p(C_1^t, C_2^t | C^{t-1}, I_1^t = 1, I_2^t = 0) \quad (15)$$

$$p(C_1^t, C_2^t | C^{t-1}, I_1^t = 1, I_2^t = 1) \neq p(C_1^t, C_2^t | C^{t-1}, I_1^t = 0, I_2^t = 1) \quad (16)$$

With these, there are two relations yet to show.

Joint Interactions vs Observational First, consider the joint interaction ($I_1^t = I_2^t = 1$) versus the pure observational regime ($I_1^t = I_2^t = 0$). We prove that these two distributions must be different by contradiction. We first assume that they are equal and show that a contradiction strictly follows. With both equations equal, we can write:

$$p(C_1^t | C^{t-1}, I_1^t = 0) \cdot p(C_2^t | C^{t-1}, I_2^t = 0) = p(C_1^t | C^{t-1}, I_1^t = 1) \cdot p(C_2^t | C^{t-1}, I_2^t = 1) \quad (17)$$

$$\Leftrightarrow \frac{p(C_1^t | C^{t-1}, I_1^t = 0)}{p(C_1^t | C^{t-1}, I_1^t = 1)} = \frac{p(C_2^t | C^{t-1}, I_2^t = 1)}{p(C_2^t | C^{t-1}, I_2^t = 0)} \quad (18)$$

Note that the third step is possible since $p(C_i^t | C^{t-1}, I_i^t = 1), p(C_i^t | C^{t-1}, I_i^t = 0)$ share the same support. Further, since C_1^t and C_2^t are conditionally independent, the equality above must hold for any values of C_1^t, C_2^t . This implies that, for a given C_2^t , the fraction of C_1^t must be constant, and vice versa. Denoting this constant factor with c , we can rewrite the previous

equation as:

$$\frac{p(C_1^t | C^{t-1}, I_1^t = 0)}{p(C_1^t | C^{t-1}, I_1^t = 1)} = c \quad (19)$$

$$\Leftrightarrow p(C_1^t | C^{t-1}, I_1^t = 0) = c \cdot p(C_1^t | C^{t-1}, I_1^t = 1) \quad (20)$$

$$\Leftrightarrow \int p(C_1^t | C^{t-1}, I_1^t = 0) dC_1^t = \int c \cdot p(C_1^t | C^{t-1}, I_1^t = 1) dC_1^t \quad (21)$$

$$\Leftrightarrow 1 = c \quad (22)$$

In the last step, the two integrals disappear since both $p(C_1^t | C^{t-1}, I_1^t = 0)$ and $p(C_1^t | C^{t-1}, I_1^t = 1)$ are valid probability density functions. Hence, the equality can only be valid if $c = 1$, which implies $p(C_1^t | C^{t-1}, I_1^t = 1) = p(C_1^t | C^{t-1}, I_1^t = 0)$. However, this equality of distributions is out ruled by the assumptions of Theorem 3.3 as discussed in the beginning of this section, and thus causes a contradiction. In other words, this shows that the joint interaction ($I_1^t = I_2^t = 1$) and the pure observational regime ($I_1^t = I_2^t = 0$) must be strictly different, *i.e.*:

$$p(C_1^t, C_2^t | C^{t-1}, I_1^t = 0, I_2^t = 0) \neq p(C_1^t, C_2^t | C^{t-1}, I_1^t = 1, I_2^t = 1) \quad (23)$$

Single-target vs Single-target The final step is to show that the distribution for interacting on C_1 versus the distribution of interacting on C_2 must be different. For this, we can use a similar strategy as for the previous comparison and perform a proof of contradiction. If both of the distributions are equal, the following equation follows:

$$\Leftrightarrow p(C_1^t | C^{t-1}, I_1^t = 0) \cdot p(C_2^t | C^{t-1}, I_2^t = 1) = p(C_1^t | C^{t-1}, I_1^t = 1) \cdot p(C_2^t | C^{t-1}, I_2^t = 0) \quad (24)$$

$$\Leftrightarrow \frac{p(C_1^t | C^{t-1}, I_1^t = 0)}{p(C_1^t | C^{t-1}, I_1^t = 1)} = \frac{p(C_2^t | C^{t-1}, I_2^t = 0)}{p(C_2^t | C^{t-1}, I_2^t = 1)} \quad (25)$$

This is almost identical to Equation (18), besides the flipped fraction for C_2 . Note, however, that the same implications hold, namely that both fractions need to be constant and constant with value 1. This again contradicts our assumptions, and proves that the two distributions must be different:

$$p(C_1^t, C_2^t | C^{t-1}, I_1^t = 0, I_2^t = 1) \neq p(C_1^t, C_2^t | C^{t-1}, I_1^t = 1, I_2^t = 0) \quad (26)$$

Conclusion In summary, we have shown that the four possible cases of interactions strictly model different distributions. Further, this distinction between the four cases can only be obtained by information from R^t through I^t , since I^t cannot be a deterministic function of the previous time step. Thus, any possible representation of the variables C_1, C_2 must model the same four (or at least three) possible interaction settings.

B.2.3 Any representation requires the same interaction cases - multi-variable case

So far, we have discussed the interaction cases for two variables. This discussion can be easily extended to cases of three or more variables. Before doing so, we formally state the lemma we are proving in this step.

Lemma B.2. *For the interaction variables $I^t = \{I_1^t, \dots, I_K^t\}$ in a causal model \mathcal{M} , any two values $a^I, b^I \in \{0, 1\}^K$ with $a^I \neq b^I$ must strictly model different distributions:*

$$\forall C^t : p(C^t | C^{t-1}, I^t = a^I) \neq p(C^t | C^{t-1}, I^t = b^I)$$

Proof. For K variables, we can write the overall joint distribution $p(C^t | C^{t-1}, I^t)$ as:

$$p(C^t | C^{t-1}, I^t) = p(C_1^t | C^{t-1}, I_1^t) \cdot p(C_2^t | C^{t-1}, I_2^t) \cdot \dots \cdot p(C_K^t | C^{t-1}, I_K^t) \quad (27)$$

Consider now the distributions for two different, arbitrary interaction values a^I, b^I ($a^I \neq b^I$):

$$p(C^t | C^{t-1}, I^t = a^I) \stackrel{?}{=} p(C^t | C^{t-1}, I^t = b^I) \quad (28)$$

$$p(C_1^t | C^{t-1}, I_1^t = a_1^I) \cdot \dots \cdot p(C_K^t | C^{t-1}, I_K^t = a_K^I) = p(C_1^t | C^{t-1}, I_1^t = b_1^I) \cdot \dots \cdot p(C_K^t | C^{t-1}, I_K^t = b_K^I) \quad (29)$$

We can rewrite this equation as:

$$\frac{p(C_1^t | C^{t-1}, I_1^t = a_1^I)}{p(C_1^t | C^{t-1}, I_1^t = b_1^I)} \cdot \dots \cdot \frac{p(C_K^t | C^{t-1}, I_K^t = a_K^I)}{p(C_K^t | C^{t-1}, I_K^t = b_K^I)} = 1 \quad (30)$$

Similar to our discussion on two variables, we can now analyze this equation under the situation where we keep all variables fixed up to C_i^{t+1} . This is a valid scenario since all variables are independent based on their conditioning set C^{t-1} and a^I/b^I . This implies that the fraction of C_i in Equation (30) must be equals to one divided by the multiplication of the remaining fractions, which we considered constant. With this, we have the following equation:

$$\frac{p(C_i^t | C^{t-1}, I_i^t = a_i^I)}{p(C_i^t | C^{t-1}, I_i^t = b_i^I)} = c \quad (31)$$

where c again summarizes all constant terms. As shown earlier in this section, this equality can only hold if $a_i^I = b_i^I$. In turn, this mean that Equation (28) can only be an equality if $a^I = b^I$. Hence, different interaction cases must strictly model different distributions. \square

B.2.4 Alignment of interaction variables - 2 variables

Setup In the previous section, we have proven that any representation needs to model the same interaction cases. The next step is to show that the interaction cases further need to align, *i.e.*, the interaction variables must be equivalent up to permutation and sign flips. For this, consider an alternative representation, \hat{C}_1, \hat{C}_2 , which is the result of an invertible change-of-variables operation. We denote the corresponding interaction variables by \hat{I}_1, \hat{I}_2 . Overall, we can write their probability distribution as:

$$p(C_1^t, C_2^t | C^{t-1}, I_1^t, I_2^t) = p(\hat{C}_1^t, \hat{C}_2^t | C^{t-1}, \hat{I}_1^t, \hat{I}_2^t) \cdot |\det \mathbf{J}| \quad (32)$$

$$p(C_1^t | C^{t-1}, I_1^t) \cdot p(C_2^t | C^{t-1}, I_2^t) = p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t) \cdot p(\hat{C}_2^t | C^{t-1}, \hat{I}_2^t) \cdot |\det \mathbf{J}| \quad (33)$$

For simplicity, we write the conditioning of \hat{C}_1^t, \hat{C}_2^t still in terms of C^{t-1} , since C^{t-1} and \hat{C}^{t-1} contain the same information. Further, \mathbf{J} represents the Jacobian of the invertible transformation of C_1^t, C_2^t to \hat{C}_1^t, \hat{C}_2^t .

Cases to consider Now, our goal is to show that \hat{I}_1^t, \hat{I}_2^t must be equivalent to I_1^t, I_2^t up to permutation and sign flip. As an example, consider a value of C^{t-1} under which we may have four possible values of R^t which give us the following interactions:

R	I_1	I_2	\hat{I}_1	\hat{I}_2
r_1	0	0	0	0
r_2	1	0	1	0
r_3	0	1	1	1
r_4	1	1	0	1

with $r_1, r_2, r_3, r_4 \in \mathcal{R}, r_1 \neq r_2 \neq r_3 \neq r_4$. In the notation of intervention design, one can interpret these different interactions are different *experiments*, *i.e.*, different sets of variables that are jointly intervened. We will denote them with E_1, \dots, E_4 where $E_i = [I_1^t, I_2^t]$ for a given r_i . Similarly, we will use $\hat{E}_i = [\hat{I}_1^t, \hat{I}_2^t]$ to denote the same set for the alternative representation.

In this setup, we say that I_2^t aligns with \hat{I}_2^t , since they are equal in all experiments E_1, \dots, E_4 for all values of R^t . However, I_1^t does not align with any interaction variable of \hat{C} , because $I_1^t \neq \hat{I}_1^t$ and $I_1^t \neq 1 - \hat{I}_1^t$, and same for \hat{I}_2^t . Thus, we are aiming to derive that this setup contradicts Equation (33).

Single-target vs Joint interaction We start the analysis by writing down all distributions to compare:

$$p(C_1^t | C^{t-1}, I_1^t = 0) \cdot p(C_2^t | C^{t-1}, I_2^t = 0) = p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t = 0) \cdot p(\hat{C}_2^t | C^{t-1}, \hat{I}_2^t = 0) \cdot |\det \mathbf{J}| \quad (34)$$

$$p(C_1^t | C^{t-1}, I_1^t = 1) \cdot p(C_2^t | C^{t-1}, I_2^t = 0) = p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t = 1) \cdot p(\hat{C}_2^t | C^{t-1}, \hat{I}_2^t = 0) \cdot |\det \mathbf{J}| \quad (35)$$

$$p(C_1^t | C^{t-1}, I_1^t = 0) \cdot p(C_2^t | C^{t-1}, I_2^t = 1) = p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t = 1) \cdot p(\hat{C}_2^t | C^{t-1}, \hat{I}_2^t = 1) \cdot |\det \mathbf{J}| \quad (36)$$

$$p(C_1^t | C^{t-1}, I_1^t = 1) \cdot p(C_2^t | C^{t-1}, I_2^t = 1) = p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t = 0) \cdot p(\hat{C}_2^t | C^{t-1}, \hat{I}_2^t = 1) \cdot |\det \mathbf{J}| \quad (37)$$

Our overall proof strategy is to derive relations between individual variables, *e.g.*, C_1 and \hat{C}_1 . Since the invertible transformation between C and \hat{C} must be independent of R , I and \hat{I} , the relations we derive must hold across all the experiments. By dividing the sets of equations, we obtain:

$$\text{Eq 35 / Eq 34 : } \frac{p(C_1^t | C^{t-1}, I_1^t = 1)}{p(C_1^t | C^{t-1}, I_1^t = 0)} = \frac{p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t = 1)}{p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t = 0)} \quad (38)$$

$$\text{Eq 36 / Eq 34 : } \frac{p(C_2^t | C^{t-1}, I_2^t = 1)}{p(C_2^t | C^{t-1}, I_2^t = 0)} = \frac{p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t = 1)}{p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t = 0)} \frac{p(\hat{C}_2^t | C^{t-1}, \hat{I}_2^t = 1)}{p(\hat{C}_2^t | C^{t-1}, \hat{I}_2^t = 0)} \quad (39)$$

$$\text{Eq 37 / Eq 34 : } \frac{p(C_1^t | C^{t-1}, I_1^t = 1) p(C_2^t | C^{t-1}, I_2^t = 1)}{p(C_1^t | C^{t-1}, I_1^t = 0) p(C_2^t | C^{t-1}, I_2^t = 0)} = \frac{p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t = 1)}{p(\hat{C}_2^t | C^{t-1}, \hat{I}_2^t = 0)} \quad (40)$$

Note that the Jacobian, $|\det J|$, cancels out in all distributions since it is independent of the interactions and thus identical for all equations above. As a next step, we replace \hat{C}_2 in Equation (39) with the result of Equation (40) and rearrange the terms:

$$\frac{p(C_1^t | C^{t-1}, I_1^t = 0)}{p(C_1^t | C^{t-1}, I_1^t = 1)} = \frac{p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t = 1)}{p(\hat{C}_1^t | C^{t-1}, \hat{I}_1^t = 0)} \quad (41)$$

Similarly, replacing \hat{C}_1 in Equation (38) with the new result in Equation (41), we obtain:

$$\frac{p(C_1^t | C^{t-1}, I_1^t = 1)}{p(C_1^t | C^{t-1}, I_1^t = 0)} = \frac{p(C_1^t | C^{t-1}, I_1^t = 0)}{p(C_1^t | C^{t-1}, I_1^t = 1)} \quad (42)$$

This equation can obviously only hold if both fractions are equal to 1. However, as shown in Appendix B.2.2, this contradicts our assumptions of the theorem. Thus, we have shown that the interaction variables \hat{I}_1, \hat{I}_2 cannot model the same distribution as I_1, I_2 . For the specific example of two causal variables and four experiments, it turns out that there exists no other set of interaction variables that would not align to I_1, I_2 . Hence, in this case, any other valid representation \hat{C} which fulfills Equation (33) must have interaction variables that align with the true model.

Conclusion This example is meant to communicate the general intuition behind our proof strategy for showing that the interaction variables between the true causal model \mathcal{M} and a learned representation $\hat{\mathcal{M}}$ align. We note that this example does not cover all possible models with two causal variables C_1, C_2 , since our assumptions only require $\lfloor \log_2 2 \rfloor + 2 = 3$ experiments/different values of R^t , while we considered here four for simplicity. For this smaller amount of experiments, it becomes difficult to distinguish between models that model the true interaction variables I_1, \dots, I_K and possible linear combinations of such. This can be prevented by ensuring sufficient variability either in the dynamics (condition (A) - Theorem 3.3) or over time (condition (B) - Theorem 3.3), which we show in the next two subsections.

B.2.5 Alignment of interaction variables - Multi-variable case (condition (A) - Theorem 3.3)

We start with showing the interaction variable alignment under condition (A) of Theorem 3.3. The goal is to prove the following lemma:

Lemma B.3 (Dynamics Variability). *For any variable C_k ($k = 1, \dots, K$) with interaction variable I_k , there exist exactly one variable \hat{C}_l with interaction variable \hat{I}_l , which models the same interaction pattern:*

$$\forall C^t : I_k^t = \hat{I}_l^t \text{ or } I_k^t = 1 - \hat{I}_l^t$$

if the second derivative of the log-difference between the observational and the interactional distribution is not constantly zero:

$$\forall C_k^t, \exists C^{t-1} : \frac{\partial^2 \Delta(C_k^t | C^{t-1})}{\partial (C_k^t)^2} \neq 0$$

Proof. We structure the proof in four main steps. First, we generalize our analysis of the relations between interaction equations from the two variables to the multi-variable case. We then take a closer look at them from two sides: a variable from the true causal model, C_m^t , and a variable from the alternative representation, \hat{C}_l^t . The intuition behind the proof is that a change in C_m^t must correspond to a change in \hat{C}_l^t which appears in the same set of equations. This inherently requires that

C_m^t and \hat{C}_l^t share the same unique interaction pattern. With this intuition in mind, the following paragraphs detail these individual proof steps.

Equations sets implied by interactions Firstly, we consider a set of Q true interaction experiments E_1, \dots, E_Q , i.e., Q different values of R^t which cause different sets of interaction variable values I_1^t, \dots, I_K^t , and similarly the Q values of R^t in the alternative representation space \hat{I} with experiments $\hat{E}_1, \dots, \hat{E}_Q$. In the previous example of the two variables, the experiments would be $E_1 = [0, 0], E_2 = [1, 0], E_3 = [0, 1], E_4 = [1, 1]$ and $\hat{E}_1 = [0, 0], \hat{E}_2 = [1, 0], \hat{E}_3 = [1, 1], \hat{E}_4 = [0, 1]$. We will denote the interaction variable I_k of the causal variable C_k in the experiment E_i with E_i^k , i.e., $E_2^1 = 1$ in the previous example.

For any two experiments E_i, E_j , there exists a set of variables for which the interaction targets differ. We summarize the indices of these variables as \mathcal{V}_{ij} , and similarly for the alternative representation $\hat{\mathcal{V}}_{ij}$. Taking the two-variable example again, $\mathcal{V}_{12} = \{1\}$, i.e., the interaction variable of the causal variable C_1 differs between E_1 and E_2 . Using this notation, we can write the division of two experiments E_i, E_j as:

$$\prod_{k \in \mathcal{V}_{ij}} \frac{p(C_k^t | C^{t-1}, I_k^t = E_i^k)}{p(C_k^t | C^{t-1}, I_k^t = E_j^k)} = \prod_{l \in \hat{\mathcal{V}}_{ij}} \frac{p(\hat{C}_l^t | C^{t-1}, \hat{I}_l^t = \hat{E}_i^l)}{p(\hat{C}_l^t | C^{t-1}, \hat{I}_l^t = \hat{E}_j^l)} \quad (43)$$

Analyzing equations for individual causal variables The experiments imply a set of $\frac{(Q-1)(Q-2)}{2}$ equations. Our next step is to analyze what these equations imply for an individual causal variable C_m^t . First, we take the log on both sides to obtain:

$$\sum_{k \in \mathcal{V}_{ij}} \log \frac{p(C_k^t | C^{t-1}, I_k^t = E_i^k)}{p(C_k^t | C^{t-1}, I_k^t = E_j^k)} = \sum_{l \in \hat{\mathcal{V}}_{ij}} \log \frac{p(\hat{C}_l^t | C^{t-1}, \hat{I}_l^t = \hat{E}_i^l)}{p(\hat{C}_l^t | C^{t-1}, \hat{I}_l^t = \hat{E}_j^l)} \quad (44)$$

For readability, we adapt our notation of $\Delta(C_k^t | C^{t-1})$ here by having:

$$\Delta_{ij}(C_k^t | C^{t-1}) = \log \frac{p(C_k^t | C^{t-1}, I_k^t = E_i^k)}{p(C_k^t | C^{t-1}, I_k^t = E_j^k)} \quad (45)$$

$$\Delta_{ij}(\hat{C}_l^t | C^{t-1}) = \log \frac{p(\hat{C}_l^t | C^{t-1}, \hat{I}_l^t = \hat{E}_i^l)}{p(\hat{C}_l^t | C^{t-1}, \hat{I}_l^t = \hat{E}_j^l)} \quad (46)$$

which gives us

$$\sum_{k \in \mathcal{V}_{ij}} \Delta_{ij}(C_k^t | C^{t-1}) = \sum_{l \in \hat{\mathcal{V}}_{ij}} \Delta_{ij}(\hat{C}_l^t | C^{t-1}) \quad (47)$$

Now consider a single variable C_m^t , for which $m \in \mathcal{V}_{ij}$. If we take the derivative with respect to C_m^t , we get:

$$\frac{\partial \Delta_{ij}(C_m^t | C^{t-1})}{\partial C_m^t} = \sum_{l \in \hat{\mathcal{V}}_{ij}} \frac{\partial \Delta_{ij}(\hat{C}_l^t | C^{t-1})}{\partial C_m^t} \quad (48)$$

The sum on the left drops away since we know that $C_k^t \perp\!\!\!\perp C_m^t | C^{t-1}, I^t$, and therefore $\frac{\partial \Delta_{ij}(C_k^t | C^{t-1})}{\partial C_m^t} = 0$ if $k \neq m$.

For each variable C_m^t , we obtain at least $Q - 1$ equations (Q being the number of overall interaction experiments) since every experiment E_i must have at least one experiment E_j for which $E_i^m \neq E_j^m$, since otherwise the interaction variable I_m must be equal in all experiments and thus a constant, violating our distinct interaction pattern assumption. In other words, we obtain a set of experiment pairs which differ in the interaction variable of C_m^t , i.e., $\mathcal{V}_m = \{\hat{\mathcal{V}}_{ij} \mid i, j \in \llbracket 1..Q \rrbracket, E_i^m \neq E_j^m\}$ with $|\mathcal{V}_m| \geq Q - 1$.

For two experiment equations, $\hat{\mathcal{V}}_{ij}, \hat{\mathcal{V}}_{sr} \in \mathcal{V}_m$, we have the following equality following from Equation (48):

$$\frac{\partial \Delta_{ij}(C_m^t | C^{t-1})}{\partial C_m^t} = \sum_{l \in \hat{\mathcal{V}}_{ij}} \frac{\partial \Delta_{ij}(\hat{C}_l^t | C^{t-1})}{\partial C_m^t} = \sum_{w \in \hat{\mathcal{V}}_{sr}} \frac{\partial \Delta_{sr}(\hat{C}_w^t | C^{t-1})}{\partial C_m^t} \quad (49)$$

Using $\Delta_{ij}(C_k^t | C^{t-1}) = -\Delta_{ji}(C_k^t | C^{t-1})$, we can align the equations above via:

$$\Delta(C_k^t | C^{t-1}) = \frac{p(C_k^t | C^{t-1}, I_k^t = 1)}{p(C_k^t | C^{t-1}, I_k^t = 0)} \quad (50)$$

$$\sum_{l \in \hat{\mathcal{V}}_{ij}} \frac{\partial \Delta(\hat{C}_l^t | C^{t-1})}{\partial C_m^t} = (-1)^{\mathbb{1}[E_i^m = E_s^m]} \sum_{w \in \hat{\mathcal{V}}_{sr}} \frac{\partial \Delta(\hat{C}_w^t | C^{t-1})}{\partial C_m^t} \quad (51)$$

Analyzing equations for a single variable of alternative representation As the next step, we analyze the derivatives of individual variables of the alternative representation \hat{C} in Equation (51). Consider a variable $\hat{C}_l^t, l \in \hat{\mathcal{V}}_{ij}$. Taking the derivative of Equation (51) with respect to \hat{C}_l^t , the left-hand side simplifies to only the \hat{C}_l^t since for all other variables, we have that $\hat{C}_l^t \perp\!\!\!\perp \hat{C}_{l'}^t | C^{t-1}, \hat{I}^t$. The right-hand side, however, has two options:

$$\frac{\partial^2 \Delta(C_m^t | C^{t-1})}{\partial C_m^t \partial \hat{C}_l^t} = \frac{\partial^2 \Delta(\hat{C}_l^t | C^{t-1})}{\partial C_m^t \partial \hat{C}_l^t} = (-1)^{\mathbb{1}[E_i^m = E_s^m]} \sum_{w \in \hat{\mathcal{V}}_{sr}} \begin{cases} 0 & \text{if } w \neq l \\ \frac{\partial^2 \Delta(\hat{C}_l^t | C^{t-1})}{\partial C_m^t \partial \hat{C}_l^t} & \text{if } w = l \end{cases} \quad (52)$$

$$= \begin{cases} 0 & \text{if } l \notin \hat{\mathcal{V}}_{sr} \\ (-1)^{\mathbb{1}[E_i^m = E_s^m]} \frac{\partial^2 \Delta(\hat{C}_l^t | C^{t-1})}{\partial C_m^t \partial \hat{C}_l^t} & \text{if } l \in \hat{\mathcal{V}}_{sr} \end{cases} \quad (53)$$

If $E_i^m \neq E_s^m$, we have an equation similar to $c = -c$, which can only be solved via $c = 0$. Therefore, we can further simplify the equation to:

$$\frac{\partial^2 \Delta(\hat{C}_l^t | C^{t-1})}{\partial C_m^t \partial \hat{C}_l^t} = \begin{cases} 0 & \text{if } l \notin \hat{\mathcal{V}}_{sr} \text{ or } E_i^m \neq E_s^m \\ \frac{\partial^2 \Delta(\hat{C}_l^t | C^{t-1})}{\partial C_m^t \partial \hat{C}_l^t} & \text{otherwise} \end{cases} \quad (54)$$

Plugging everything together From Equation (54), we can make the following conclusions: for any variable \hat{C}_l^t which is not in *all* experiment pairs of \mathcal{V}_m , its second derivative $\frac{\partial^2 \Delta(\hat{C}_l^t | C^{t-1})}{\partial C_m^t \partial \hat{C}_l^t}$ must be zero. This is an important insight, since we know that all second derivatives equations must still equal to $\frac{\partial^2 \Delta(C_m^t | C^{t-1})}{\partial C_m^t \partial \hat{C}_l^t}$. Using the chain rule, we can relate these second derivatives even further:

$$\frac{\partial^2 \Delta(C_m^t | C^{t-1})}{\partial C_m^t \partial \hat{C}_l^t} = \frac{\partial^2 \Delta(C_m^t | C^{t-1})}{\partial^2 C_m^t} \mathbf{J}_{ml}^{-1} \quad (55)$$

$$\frac{\partial^2 \Delta(C_m^t | C^{t-1})}{\partial^2 C_m^t} \mathbf{J}_{ml}^{-1} = \frac{\partial^2 \Delta(\hat{C}_l^t | C^{t-1})}{\partial C_m^t \partial \hat{C}_l^t} \quad (56)$$

where \mathbf{J}_{ml}^{-1} is the ml -th entry of the inverse of the Jacobian, *i.e.*, $\mathbf{J}_{ml}^{-1} = \frac{\partial C_m^t}{\partial \hat{C}_l^t}$. From our assumptions, we know that $\frac{\partial^2 \Delta(C_m^t | C^{t-1})}{\partial^2 C_m^t}$ cannot be constant zero for all values of C_m^t . Therefore, if $\frac{\partial^2 \Delta(\hat{C}_l^t | C^{t-1})}{\partial C_m^t \partial \hat{C}_l^t}$ is zero following Equation (54), then this must strictly imply that \mathbf{J}_{ml}^{-1} must be constantly zero, *i.e.*, C_m^t and \hat{C}_l^t are independent.

However, at the same time, we know that \mathbf{J}_{ml}^{-1} cannot be constantly zero for all l since otherwise, \mathbf{J}^{-1} (and therefore \mathbf{J}) has a zero determinant and thus the transformation between C and \hat{C} cannot be invertible. Therefore, in order for \hat{C} to be a valid transformation, there must exist at least one variable \hat{C}_l which is in all experiment sets \mathcal{V}_m . This implies that for this variable \hat{C}_l and our original causal variable C_m , the following relations must hold:

$$E_i^k = E_j^k \Leftrightarrow \hat{E}_i^l = \hat{E}_j^l \quad (57)$$

This inherently implies that for any variable C_k , there must exist at least one variable \hat{C}_l , for which the following must hold:

$$\forall i, E_i^k = \hat{E}_i^l \quad \text{or} \quad \forall i, E_i^k = 1 - \hat{E}_i^l \quad (58)$$

Finally, since for every variable C_k , the set of experiments is unique, *i.e.*, no deterministic function between I_k and any other interaction variable I_j , and the alternative representation has the same number of variables, it implies that there exists a 1-to-1 match between an interaction variable I_k and in the alternative representation \hat{I}_k . This proves our original lemma. \square

B.2.6 Alignment of interaction variables - Multi-variable case (condition (B) - Theorem 3.3)

Lemma B.4 (Time-variability). *For any variable C_k ($k = 1, \dots, K$) with interaction variable I_k , there exist exactly one variable \hat{C}_l with interaction variable \hat{I}_l , which models the same interaction pattern:*

$$\forall C^t : I_k^t = \hat{I}_l^t \text{ or } I_k^t = 1 - \hat{I}_l^t$$

if for any C^t , there exist $K + 1$ different values c^1, \dots, c^{K+1} for C^t for which the vectors v_1, \dots, v_K of the following structure are linearly independent:

$$v_i = \left[\frac{\partial \Delta(C_i^t | C^{t-1} = c^1)}{\partial C_i^t} \quad \frac{\partial \Delta(C_i^t | C^{t-1} = c^2)}{\partial C_i^t} \quad \dots \quad \frac{\partial \Delta(C_i^t | C^{t-1} = c^{K+1})}{\partial C_i^t} \right]^T \in \mathbb{R}^K$$

Proof. We follow the same proof as for Lemma B.3 up until Equation (48), where for each variable C_m , we have obtained the following equation:

$$\frac{\partial \Delta_{ij}(C_m^t | C^{t-1})}{\partial C_m^t} = \sum_{l \in \hat{\mathcal{V}}_{ij}} \frac{\partial \Delta_{ij}(\hat{C}_l^t | C^{t-1})}{\partial C_m^t} \quad (59)$$

Here, we rewrite the derivative $\frac{\partial \Delta_{ij}(\hat{C}_l^t | C^{t-1})}{\partial C_m^t}$ using the chain rule to:

$$\frac{\partial \Delta_{ij}(\hat{C}_l^t | C^{t-1})}{\partial C_m^t} = \frac{\partial \Delta_{ij}(\hat{C}_l^t | C^{t-1})}{\partial \hat{C}_l^t} \frac{\partial \hat{C}_l^t}{\partial C_m^t} \quad (60)$$

$$= \frac{\partial \Delta_{ij}(\hat{C}_l^t | C^{t-1})}{\partial \hat{C}_l^t} \mathbf{J}_{lm} \quad (61)$$

Therefore, we obtain:

$$\frac{\partial \Delta_{ij}(C_m^t | C^{t-1})}{\partial C_m^t} = \sum_{l \in \hat{\mathcal{V}}_{ij}} \frac{\partial \Delta_{ij}(\hat{C}_l^t | C^{t-1})}{\partial \hat{C}_l^t} \mathbf{J}_{lm} \quad (62)$$

Note hereby that \mathbf{J}_{lm} is independent of the time index t and particularly C^{t-1} , which will become important in the next steps of the proof.

Alternative representation having linear independent vectors Now consider the K different vectors v^1, \dots, v^K , which are linearly independent. For each of these individual vectors, we have at least K equations of the form of Equation (59), namely for each C_1^t, \dots, C_K^t . We can also express this in the form of a matrix product. For that, we first stack the vectors v^1, \dots, v^K into $\mathbf{V} \in \mathbb{R}^{(K+1) \times K}$:

$$\mathbf{V} = \begin{bmatrix} | & & | \\ v_1 & \cdots & v_K \\ | & & | \end{bmatrix} = \begin{bmatrix} \frac{\partial \Delta(C_1^t | C^{t-1} = c^1)}{\partial C_1^t} & \frac{\partial \Delta(C_2^t | C^{t-1} = c^1)}{\partial C_2^t} & \dots & \frac{\partial \Delta(C_K^t | C^{t-1} = c^1)}{\partial C_K^t} \\ \frac{\partial \Delta(C_1^t | C^{t-1} = c^2)}{\partial C_1^t} & \frac{\partial \Delta(C_2^t | C^{t-1} = c^2)}{\partial C_2^t} & \dots & \frac{\partial \Delta(C_K^t | C^{t-1} = c^2)}{\partial C_K^t} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \Delta(C_1^t | C^{t-1} = c^{K+1})}{\partial C_1^t} & \frac{\partial \Delta(C_2^t | C^{t-1} = c^{K+1})}{\partial C_2^t} & \dots & \frac{\partial \Delta(C_K^t | C^{t-1} = c^{K+1})}{\partial C_K^t} \end{bmatrix} \quad (63)$$

We denote $\hat{\mathbf{V}}$ as the same matrix as in Equation (63), just with each C_i^t replaced with \hat{C}_i^t . Finally, we need to represent the factors of Equation (62). Since these depend on a specific pair of experiments E_i, E_j , we pick for each variable C_m^{t+1} an arbitrary pair of experiments, where $E_i^m = 1$ and $E_j^m = 0$. With this in mind, we can express the factors of Equation (62) as:

$$\mathbf{L} = \mathbf{J} \odot \begin{bmatrix} \delta_{11}^E & \delta_{12}^E & \dots & \delta_{1K}^E \\ \delta_{21}^E & \delta_{22}^E & \dots & \delta_{2K}^E \\ \vdots & \vdots & \ddots & \vdots \\ \delta_{K1}^E & \delta_{K2}^E & \dots & \delta_{KK}^E \end{bmatrix} \quad (64)$$

where $\mathbf{L} \in \mathbb{R}^{K \times K}$, \odot is the Hamard product/element-wise product, and $\delta_{lm}^E = 1$ if $\hat{E}_i^l = 1, \hat{E}_j^l = 0$ for the experiment pair E_i, E_j picked for C_m , 0 if $\hat{E}_i^l = \hat{E}_j^l$, and -1 otherwise. With that, we can express Equation (62) in matrix form as:

$$\mathbf{V} = \hat{\mathbf{V}}\mathbf{L} \quad (65)$$

Since \mathbf{V} has linearly independent columns and, based on Equation (65), is equal to linear combinations of the columns of $\hat{\mathbf{V}}$, it directly follows that $\hat{\mathbf{V}}$ must also have linearly independent columns.

Solution to linear independent system At the same time, we know that for each variable C_m , there exist pairs of experiments E_i, E_j for which $E_i^m = E_j^m$. We denote this set of experiment pairs by $\bar{\mathcal{V}}^m = \{\bar{\mathcal{V}}_{ij} \mid i, j \in [1..Q], E_i^m = E_j^m\}$ with its size denoted as $Q_m = |\bar{\mathcal{V}}^m|$. Each of these implies an equation like the following:

$$0 = \sum_{l \in \bar{\mathcal{V}}_{ij}} \frac{\partial \Delta_{ij}(\hat{C}_l^t | C^{t-1})}{\partial \hat{C}_l^t} \mathbf{J}_{lm} \quad (66)$$

We can, again, write it in matrix form to show this set of equations over the different temporal values c^1, \dots, c^{K+1} :

$$\mathbf{0} = \hat{\mathbf{V}} \bar{\mathbf{L}}^m \quad (67)$$

where

$$\bar{\mathbf{L}}^m = \begin{bmatrix} \mathbf{J}_{.m} & \cdot & \mathbf{J}_{.m} \\ \vdots & & \vdots \end{bmatrix} \odot \begin{bmatrix} \mathbb{1}[1 \in \bar{\mathcal{V}}_1^m] & \mathbb{1}[1 \in \bar{\mathcal{V}}_2^m] & \cdots & \mathbb{1}[1 \in \bar{\mathcal{V}}_{Q_m}^m] \\ \mathbb{1}[2 \in \bar{\mathcal{V}}_1^m] & \mathbb{1}[2 \in \bar{\mathcal{V}}_2^m] & \cdots & \mathbb{1}[2 \in \bar{\mathcal{V}}_{Q_m}^m] \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{1}[K \in \bar{\mathcal{V}}_1^m] & \mathbb{1}[K \in \bar{\mathcal{V}}_2^m] & \cdots & \mathbb{1}[K \in \bar{\mathcal{V}}_{Q_m}^m] \end{bmatrix} \quad (68)$$

Intuitively, $\bar{\mathbf{L}}^m$ lists out the Q_m different equations of Equation (66) for variable C_m , duplicated for all possible temporal values c^1, \dots, c^{K+1} . Since all columns of $\hat{\mathbf{V}}$ are linearly independent, the only solution to the system is that $\bar{\mathbf{L}}^m = \mathbf{0}$, or in index form $\bar{\mathbf{L}}_{ij}^m = 0$ for all $i = 1, \dots, Q_m; j = 1, \dots, K$.

Matching of interaction variables The fact that $\bar{\mathbf{L}}_{ij}^m = 0$ must be zero means that one of its two matrix elements must have a zero entry. Thus, for each variable \hat{C}_l , \mathbf{J}_{lm} can only be non-zero if l is not in any sets of $\bar{\mathcal{V}}^m$. This implies that either \hat{C}_l must follow the exact same interaction pattern as C_m , *i.e.*, $I_m = \hat{I}_l$ or $I_m = 1 - \hat{I}_l$, or \hat{I}_l is a constant value. However, the constant value case can directly be excluded since this would imply \mathbf{L} to have a zero determinant ($\delta_l^E = 0$), which is not possible with $\det \mathbf{V} \neq 0$. At the same time, at least one value of $\mathbf{J}_{.m}$ must be non-zero to ensure \mathbf{J} to be invertible. Therefore, each variable C_m must have one variable \hat{C}_l for which $I_m = \hat{I}_l$ or $I_m = 1 - \hat{I}_l$. Finally, this match of C_m, \hat{C}_l must be a unique since every variable C_m has a different interaction pattern, and we are limited to K variables $\hat{C}_1, \dots, \hat{C}_K$. With that, we have proven the initial lemma.

Non-zero elements in Jacobian Additionally to the lemma, this proof also shows that \mathbf{J} must have exactly one non-zero value in each column and row, *i.e.*, being a permuted diagonal matrix. \square

B.2.7 Equivalence up to component-wise invertible transformations and permutation

With both Lemma B.3 and Lemma B.4, we have shown that the two representations C and \hat{C} need to have the same interaction patterns. Now, we are ready to prove the identifiability of the individual causal variables. Since most of the results have been already shown in the previous proofs, we skip the intuition on the two-variable case and directly jump to the multi-variable case:

Lemma B.5. *For any variable C_k ($k = 1, \dots, K$) with interaction variable I_k , there exist exactly one variable \hat{C}_l with an invertible transformation T_l for which the following holds:*

$$C_k^t = T_l(\hat{C}_l^t)$$

Proof. We start with reiterating the initial result of Appendix B.2.1 stating that there exist an invertible transformation between C and \hat{C} : $C = T(\hat{C})$. This also gives us the change of variables distribution:

$$p(C^t | C^{t-1}, I^t) = p(\hat{C}^t | C^{t-1}, \hat{I}^t) |\det \mathbf{J}| \quad (69)$$

Our goal is to show it follows for each C_k , there exists a \hat{C}_l for which the following holds:

$$p(C_k^t | C^{t-1}, I_k^t) = p(\hat{C}_l^t | C^{t-1}, \hat{I}_l^t) |\mathbf{J}_{lk}| \quad (70)$$

This change-of-variable equation implies that there exist an invertible transformation between C_k and \hat{C}_l with the scalar Jacobian \mathbf{J}_{lk} .

Intermediate proof step based on Lemma B.3 To prove this based on Lemma B.3, we reuse our final results of the proof in Appendix B.2.5. Specifically, we have shown before that the inverse of the Jacobian \mathbf{J}_{kj}^{-1} for the transformation from C_k to \hat{C}_j must be constantly zero if C_k and \hat{C}_j do not share the same interaction pattern. Further, we have shown that for each variable C_k , there exists exactly one variable \hat{C}_l for which $\mathbf{J}_{kl}^{-1} \neq 0$. Given that \mathbf{J}_k^{-1} is zero except for entry l , and that this entry index is different for every k , it follows that \mathbf{J}_{kl}^{-1} must be a permuted diagonal matrix:

$$\mathbf{J}^{-1} = \mathbf{D}\mathbf{P} \quad (71)$$

where \mathbf{D} is a diagonal matrix and \mathbf{P} is a permutation matrix. The diagonal elements of \mathbf{D}^{-1} are the non-zero values of \mathbf{J}^{-1} , i.e., where $\mathbf{J}_{kl}^{-1} \neq 0$. Inverting both sides gives us:

$$\mathbf{J} = \mathbf{P}^T \mathbf{D}^{-1} \quad (72)$$

Inverting the diagonal matrix \mathbf{D} gives us yet another diagonal matrix, just with inverted values. Therefore, we have that $\mathbf{J}_{kl} = \frac{1}{\mathbf{J}_{lk}^{-1}}$ if $\mathbf{J}_{lk}^{-1} \neq 0$, and 0 otherwise.

Intermediate proof step based on Lemma B.4 In the proof of Lemma B.4 (Appendix B.2.6), we have already shown that \mathbf{J} must be a permuted diagonal matrix.

Joint final step With having \mathbf{J} identified as a permuted diagonal matrix, we can derive the originally stated component-wise invertible transformation. For clarity, we denote the indices at which the Jacobian is non-zero as $f(l) = \arg \max_{(l,k)} |\mathbf{J}_{lk}|$, i.e., $f(l)$ returns the index (l, k) for which $\mathbf{J}_{lk} \neq 0$. Using these indices, we can write the determinant of \mathbf{J} as the product of the individual diagonal elements:

$$|\det \mathbf{J}| = \prod_{l=1}^K |\mathbf{J}_{f(l)}| \quad (73)$$

Inherently, we can use this to rewrite Equation (69) to:

$$p(C^t | C^{t-1}, I^t) = p(\hat{C}^t | C^{t-1}, \hat{I}^t) \prod_{l=1}^K |\mathbf{J}_{f(l)}| \quad (74)$$

$$= \prod_{l=1}^K p(\hat{C}_l^t | C^{t-1}, \hat{I}_l^t) |\mathbf{J}_{f(l)}| \quad (75)$$

Therefore, for a pair of variables C_k, \hat{C}_l with $f(l) = (l, k)$, it follows that:

$$p(C_k^t | C^{t-1}, I_k^t) = p(\hat{C}_l^t | C^{t-1}, \hat{I}_l^t) |\mathbf{J}_{lk}| \quad (76)$$

This shows that for every variable C_k , there exist one variable \hat{C}_l with an invertible transformation $C_k^t = T_l(\hat{C}_l^t)$ which has the Jacobian of $|\mathbf{J}_{lk}|$. \square

B.2.8 Putting everything together

Having proven Lemma B.1, B.2, B.3, B.4, and B.5, we have now all components to prove the original theorem:

Theorem B.6. *An estimated model $\widehat{\mathcal{M}} = \langle \widehat{g}, \widehat{f}, \widehat{\omega}, \widehat{\mathcal{C}} \rangle$ identifies the true causal model $\mathcal{M} = \langle g, f, \omega, \mathcal{C} \rangle$ if:*

1. *(Observations) $\widehat{\mathcal{M}}$ and \mathcal{M} model the same likelihood:*

$$p_{\widehat{\mathcal{M}}}(X^t | X^{t-1}, R^t) = p_{\mathcal{M}}(X^t | X^{t-1}, R^t);$$

2. *(Distinct Interaction Patterns) Each variable C_i in \mathcal{M} has a distinct interaction pattern (Definition 3.2);*

and one of the following two conditions holds for \mathcal{M} :

- A. *(Dynamics Variability) Each variable's log-likelihood difference is twice differentiable and not always zero:*

$$\forall C_i^t, \exists C^{t-1}: \frac{\partial^2 \Delta(C_i^t | C^{t-1})}{\partial (C_i^t)^2} \neq 0;$$

- B. *(Time Variability) For any $C^t \in \mathcal{C}$, there exist $K + 1$ different values of C^{t-1} denoted with $c^1, \dots, c^{K+1} \in \mathcal{C}$, for which the vectors $v_1, \dots, v_K \in \mathbb{R}^{K+1}$ with*

$$v_i = \left[\frac{\partial \Delta(C_i^t | C^{t-1} = c^1)}{\partial C_i^t} \quad \dots \quad \frac{\partial \Delta(C_i^t | C^{t-1} = c^{K+1})}{\partial C_i^t} \right]^T$$

are linearly independent.

Proof. Based on Lemma B.1, we have shown that there exists an invertible transformation between the latent spaces of \mathcal{M} and $\widehat{\mathcal{M}}$. Further, we have shown in Lemma B.2 with Lemma B.3 (for condition (A)) or Lemma B.4 (for condition (B)) that $\widehat{\mathcal{M}}$ must model the same interaction cases and patterns as \mathcal{M} . Finally, this resulted in the proof of Lemma B.5, namely that the invertible transformation T has a Jacobian with the structure of a permuted diagonal matrix. This shows that there exist component-wise invertible transformations between the latent spaces of \mathcal{M} and $\widehat{\mathcal{M}}$, effectively identifying the causal variables of \mathcal{M} . \square

B.3 EXTENSION TO LONGER TEMPORAL DEPENDENCIES

The shown proof demonstrates the identifiability results for causal relations between C^t and its previous time step C^{t-1} . In case the ground truth system contains longer temporal dependencies, e.g. $C^{t-\tau} \rightarrow C^t$ for any $\tau > 1$, we can easily obtain the same identifiability results by extending our conditioning set of the distribution $p(C^t | C^{t-1}, R^t)$ to include $\tau \geq 1$ additional time steps $p(C^t | C^{t-1}, C^{t-2}, \dots, C^{t-\tau}, R^t)$.

The key property of the identifiability proof in Appendix B, that allows for this simple extension, is that we use C^{t-1} only to ensure that the causal variables in a time step t remain conditionally independent given R^t :

$$C_i^t \perp\!\!\!\perp C_j^t \mid C^{t-1}, R^t \tag{77}$$

In order to extend this to longer dependencies up to $t - \tau$, we can instead consider:

$$C_i^t \perp\!\!\!\perp C_j^t \mid C^{t-1}, \dots, C^{t-\tau}, R^t \tag{78}$$

Furthermore, the conditioning set can also be extended with any other observable information, e.g. environment parameters, as long as the conditional independencies hold. In the proof, this corresponds to replacing C^{t-1} with the set of time steps that may have a causal relation to C^t , $\{C^{t-\tau} | \tau = 1, \dots, T_D\}$, with T_D denoting the maximal temporal length of causal relations.

Our learning algorithm, BISCUIT, can be similarly adapted to longer temporal relations. Specifically, we need to condition its prior p_ω and interaction MLP $\text{MLP}_\omega^{\widehat{f}_i}$ on more time steps, i.e. changing $p_\omega(z^t | z^{t-1}, R^t)$ and $\text{MLP}_\omega^{\widehat{f}_i}(R^t, z^{t-1})$ to $p_\omega(z^t | z^{t-1}, z^{t-2}, \dots, z^{t-\tau}, R^t)$ and $\text{MLP}_\omega^{\widehat{f}_i}(R^t, z^{t-1}, \dots, z^{t-\tau})$.

Table 3: Overview of the setups under which identifiability results have been derived in causal representation learning. As additional observations, we distinguish between counterfactuals, intervention targets, and regime variable. Further, we show which causal relations (over time and/or instantaneous) are supported by the respective identifiability results. BISCUIT shares a similar setup with iVAE, LEAP and DMS.

Method	Observations / Inputs			Causal Relations	
	Counterfactuals	Intervention targets	Regime	Temporal	Instantaneous
Locatello et al. (2020)	✓	✗	✗	✗	✗
LCM (Brehmer et al., 2022)	✓	✗	✗	✗	✓
Ahuja et al. (2022)	✓	✗	(✓)	✗	✗
CITRIS (Lippe et al., 2022b)	✗	✓	✗	✓	✗
iCITRIS (Lippe et al., 2023)	✗	✓	✗	✓	✓
iVAE (Khemakhem et al., 2020a)	✗	✗	✓	(✓)	✗
LEAP (Yao et al., 2022b)	✗	✗	✓	✓	✗
DMS (Lachapelle et al., 2022a)	✗	✗	✓	✓	✗
BISCUIT (ours)	✗	✗	✓	✓	✗

B.4 IDENTIFYING THE TEMPORAL CAUSAL GRAPH

In our setting, we assume that the relations between underlying causal variables are limited to edges in a causal graph that go from a variable C_i^{t-1} at time step $t - 1$ to another variable C_j^t in the following time step t . As summarized in Figure 2, the edges between the interaction variables I_i^t and the relevant causal variables C_i^t are fixed, as are the edges between the regime R^t and the interaction variables I_i^t .

If the true causal variables were observed, by additionally assuming also the standard causal Markov and faithfulness assumptions (which are not otherwise necessary in our setup), one could easily learn the causal relations between C_i^{t-1} and C_j^t by checking which of these causal variables are still dependent when conditioning on $C^{t-1} \setminus C_i^{t-1}$ and R^{t-1} . This is a trivial modification of known results for causal discovery on time series (Peters et al., 2013), proving that the causal graph in this setting is identifiable.

In our setting, we do not know the true causal variables, but we learn the causal variables up to permutation and component-wise transformations. By applying any appropriate causal discovery algorithm for this time series setting, *e.g.*, as described by Assaad et al. (2022), we can then identify the causal structure, again up to permutation of the nodes. We provide an example of discovering the causal graph between the learned causal variables of BISCUIT in Appendix C.1.3.

B.5 RELATION TO PREVIOUS IDENTIFIABILITY RESULTS

We compare our identifiability results to various previous works in causal representation learning in terms of the different inputs/observations the methods require, and the causal relations they support in Table 3. The related works which have the most similar setups to ours, *i.e.*, using a regime variable and focuses on temporal causal relations, are iVAE (Khemakhem et al., 2020a), LEAP (Yao et al., 2022b) and DMS (Lachapelle et al., 2022b). In summary, iVAE and LEAP require a stronger form of both our dynamics and time variability assumptions, which excludes common models like additive Gaussian noise models. DMS requires that no two causal variables share the same parents, limiting the allowed temporal graph structures. We give a more detailed discussion below, which we will add in short to the main paper (Section 3 and 5) and in its full form in the appendix of our paper.

In iVAE (Khemakhem et al., 2020a), the regime variable u is assumed to contain any additional information that makes the causal variables conditionally independent. In our case, u would include both R^t and the previous time step C^{t-1} . Under this setup, the iVAE theorem states that an additive Gaussian noise model can only be identified up to a linear transformation, *e.g.* $\hat{C}^t = AC^t + c$ with $A \in \mathbb{R}^{K \times K}$, $c \in \mathbb{R}$. In comparison, the identifiability class in BISCUIT for the additive Gaussian noise model is much stronger, since the causal variables are identified up to component-wise invertible transformations. In order to gain a similar identifiability class as BISCUIT, iVAE requires the distributions $p(C_i^t | u)$ to have a sufficient statistic that is either not monotonic, or of size greater than 2 (similar to our dynamics variability assumption). Additionally, iVAE requires $2K + 1$ different regimes with linearly independent effects on the causal variables, which is similar to the linear independence in our time variability assumption.

The identifiability theorem of LEAP (Yao et al., 2022b) has similar differences to ours as iVAE. In short, it requires a stronger form of our dynamics and time variability assumption. Specifically, the theorem requires that for all regimes R^t and time steps C^t , there exist $2K + 1$ values of C^{t-1} for which the first and second-order derivative of the log-likelihood differences (in our notation $\partial\Delta(C_i^t|C^{t-1})/\partial C_i^t$ and $\partial\Delta(C_i^t|C^{t-1})^2/\partial^2 C_i^t$) are linearly independent. Thus, it requires our time variability assumption for both the first and second derivative, as well as $2K + 1$ points instead of $K + 1$ as in BISCUIT. This excludes common models such as additive Gaussian noise models.

Finally, the Disentanglement via Mechanism Sparsity method (DMS) (Lachapelle et al., 2022b) is based on the same concepts as iVAE. In order to allow for additive Gaussian noise models, the following assumptions need to be taken: (1) each causal variable has a unique set of parents, (2) the distribution functions for each causal variable need to vary sufficiently over time, and (3) there exist $K + 1$ values of C^{t-1} that change the distributions for each causal variable in a linearly independent manner (similar to our time variability assumption). The assumption of unique parents may be violated in settings where causal variables strongly interact, especially when we have context-dependent interactions (e.g. the egg being cooked by the stove, only if it is in the pan). In comparison, BISCUIT can be applied to any graph structure and does not require additional variability assumptions, making it overall wider applicable.

We note that the stronger identifiability results of BISCUIT are only possible by taking the assumption of binary interactions. In situations where the interactions between the regime variable and the causal variables cannot be described by binary variables, iVAE, LEAP or DMS may still be applicable.

B.6 DETAILED EXAMPLE FOR ADDITIVE GAUSSIAN NOISE

In this section, we discuss the additive Gaussian Noise example of Section 3.2 in detail, including the specific mechanisms used in Figure 3. Consider an additive Gaussian noise model with two variables C_1, C_2 and their dynamics function $C_i^t = \mu_i(C^{t-1}, I_i^t) + \epsilon_i, \epsilon_i \sim \mathcal{N}(0, \sigma^2)$. For simplicity, in this example we assume the regime variable can be represented as $R^t = [I_1^t, I_2^t]$ and the mean function μ_i to be of the following form:

$$\mu_i(C^{t-1}, I_i^t) = \begin{cases} 1 & \text{if } I_i^t = 1 \\ 0 & \text{otherwise} \end{cases} \quad (79)$$

A common difficulty in additive Gaussian noise models is to identify the true causal variables, C_1, C_2 , in contrast to possible rotated representations, *i.e.*, $\hat{C}_1 = \cos(\theta)C_1 - \sin(\theta)C_2, \hat{C}_2 = \sin(\theta)C_1 + \cos(\theta)C_2$ for an angle $\theta \in [0, 2\pi)$. Thus, for simplicity, we limit the possible representation space here to rotations of the form \hat{C}_1, \hat{C}_2 . We aim to show that if we enforce the dependency of the regime variable R^t with each causal variable (\hat{C}_1, \hat{C}_2) to be expressed by a binary interaction variable (\hat{I}_1, \hat{I}_2) , \hat{C}_1 and \hat{C}_2 must identify the true causal variables up to permutation and element-wise invertible transformations.

Firstly, in the general case, we can write the mean functions for the rotated representation \hat{C}_1, \hat{C}_2 as:

$$\hat{\mu}_1(\hat{C}^{t-1}, R^t) = \begin{cases} \cos(\theta) - \sin(\theta) & \text{if } R^t = [1, 1] \\ \cos(\theta) & \text{if } R^t = [1, 0] \\ -\sin(\theta) & \text{if } R^t = [0, 1] \\ 0 & \text{if } R^t = [0, 0] \end{cases} \quad (80)$$

$$\hat{\mu}_2(\hat{C}^{t-1}, R^t) = \begin{cases} \sin(\theta) + \cos(\theta) & \text{if } R^t = [1, 1] \\ \sin(\theta) & \text{if } R^t = [1, 0] \\ \cos(\theta) & \text{if } R^t = [0, 1] \\ 0 & \text{if } R^t = [0, 0] \end{cases} \quad (81)$$

Since the Gaussian distribution is rotational invariant, the fact that we can determine the mean from C^{t-1}, R^t for the new representation implies that C_1, C_2 and \hat{C}_1, \hat{C}_2 model the same data likelihood, *i.e.* $\prod_{i=1}^2 p_i(C_i^t|C^{t-1}, R^t) = \prod_{i=1}^2 \hat{p}_i(\hat{C}_i^t|\hat{C}^{t-1}, R^t)$. Therefore, without constraints on how R^t is used in the distributions for the individual causal variables, we cannot distinguish the true causal variables from an arbitrarily rotated representation simply from the data likelihood under different interactions.

This changes when making use of binary interaction variables \hat{I}_1, \hat{I}_2 . In this setting, both mean functions need to be reduced to the form:

$$\hat{\mu}_i(\hat{C}^{t-1}, \hat{I}_i^t) = \begin{cases} a(\hat{C}^{t-1}) & \text{if } \hat{I}_i^t = 1 \\ b(\hat{C}^{t-1}) & \text{otherwise} \end{cases} \quad (82)$$

where a, b can be arbitrary functions. In other words, we need to reduce the mean functions for the general setting from 4 cases to 2 cases per causal variables. Hereby, it becomes clear that we can only do so by setting either $\sin(\theta)$ or $\cos(\theta)$ to zero, which is at $\theta \in \{0, \pi/2, \pi, 3\pi/2\}$. Any of these rotations are a multiple of 90 degrees, which means that \hat{C}_1, \hat{C}_2 are identical to C_1, C_2 up to permutation and/or sign-flips. Therefore, by enforcing the effect of R^t to be described by binary interaction variables, we have successfully identified the causal variables according to Definition 3.1.

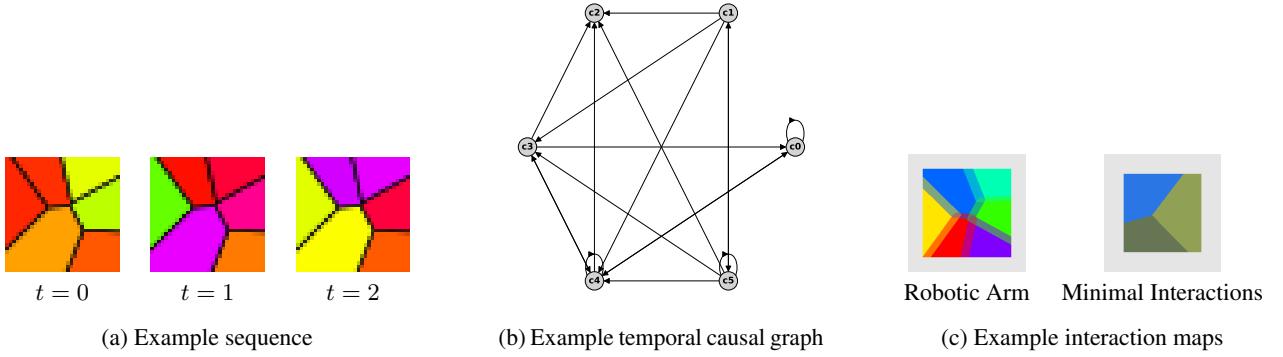


Figure 9: (a) Example sequence in the Voronoi dataset (Lippe et al., 2023) for six variables. The six colors in the images represent an entangled view of the true causal variables. (b) An example graph sampled in the Voronoi dataset for six variables. Each edge $C_i \rightarrow C_j$ represents a causal relation from C_i^{t-1} to C_j^t . (c) Example interaction maps. The two spatial dimensions represents the two dimensions of R^t , with different colors representing different interventional experiments. Gray represents the observational regime. For more examples and predictions, see Figure 10.

C EXPERIMENTAL DETAILS

In this section, we provide details on the experimental setup, including datasets and hyperparameters. Appendix C.1, C.2, and C.3 discuss the Voronoi, CausalWorld, and iTHOR experiments respectively.

C.1 VORONOI

C.1.1 Dataset Setup

The Voronoi dataset (Lippe et al., 2023) represents a synthetic benchmark for testing the model on various distributions, graph structures and variable sizes. We adapt the original setup of Lippe et al. (2023) by removing instantaneous effects and restrict the distributions to additive Gaussian noise models:

$$C_i^t = \text{MLP}_i(C^{t-1} \odot M_i) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, 0.4) \quad (83)$$

where $M_i \in \{0, 1\}^K$ is a binary mask according to the sampled causal graph (*i.e.*, $M_{ij} = 1$ if $C_j^{t-1} \rightarrow C_i^t$, and $M_{ij} = 0$ otherwise). The network MLP_i is a randomly initialized 3-layer MLP with BatchNorm layers in between. Under interventions, we set the output of the MLP to zero, effectively performing a perfect intervention while keeping the same noise distribution. For the graph structures, we sample each edge independently with a chance of 0.4, *i.e.*, $M_{ij} \sim \text{Bern}(0.4)$, while ensuring each variable having at least one parent. For a graph with six variables, this gives 2.4 parents per variable in expectation, and 3.6 for nine variables. The regime variable is a two-dimensional continuous variable in the range $R^t \in [-1.5, 1.5]^2$ and is sampled uniformly at each time step. If any dimension has an absolute value greater than 1, the data point is considered observational, *i.e.*, no interventions. For all other values in the *Robotic Arm* setup, we assign each voronoi cluster to one causal variable randomly. If a circle with radius $1/16$ around R^t touches a voronoi cluster, we perform an intervention on this respective causal variable. On boundaries, this results in performing interventions on multiple variables simultaneously. In the *Minimal Interactions* setup, we use the same strategy but limit the number of voronoi clusters to $\lfloor \log_2 K \rfloor + 2$ as in Lippe et al. (2022c), which is four for six variables and five for nine variables. Further, we do not allow for any overlap. Numerating the clusters by $c = 1, \dots, \lfloor \log_2 K \rfloor + 2$, we perform an intervention on variable C_i if $\lfloor \frac{(i+1)}{2^{c-1}} \rfloor \bmod 2 = 0$. This gives each causal variable a unique pattern in its interaction variable. An example setup for both cases is shown in Figure 9. For each dataset, we sample 150k training frames in a single sequence, and 25k samples for the held-out test set.

C.1.2 Hyperparameters and Implementation Details

The hyperparameters shared across all methods are shown in Appendix C.1.2. The CNN used for the encoder and decoder is the same as in Lippe et al. (2023). The number of latents is twice as the number of causal variables, which represents a rough overestimation of the true number of causal variables in the environment. Further, the learning rate was finetuned

Table 4: Shared hyperparameters across all methods. The learning rate was separately finetuned for each method based on the NLL loss. For individual hyperparameters, see Appendix C.1.2.

Hyperparameter	iVAE	LEAP	DMS	BISCUIT
Encoder architecture		5-layer CNN		
Decoder architecture		5-layer CNN		
Channel size		32		
Number of latents		$2 \cdot K$		
Learning rate	4e-4	4e-4	4e-4	4e-4
Optimizer		Adam (Kingma et al., 2015)		
Batch size		256		
Number of epochs		100		

separately for each method in the range [1e-4, 1e-3]. However, none of the tested methods showed to be sensitive to these hyperparameters, and 4e-4 showed to work well across all methods. In the paragraph below, we discuss further implementation details specific to the individual methods.

BISCUIT In the prior, $\text{MLP}_{\omega}^{z_i}$ is implemented by a 2-layer MLP with hidden dimensionality 32 and SiLU (Ramachandran et al., 2017) activations. $\text{MLP}_{\omega}^{\hat{f}_i}$ shares the same structure. The output activation function of $\text{MLP}_{\omega}^{\hat{f}_i}$ is implemented as $f(x) = \tanh(x \cdot \tau)$ where τ represents the inverse of the temperature and is scaled linearly in the range [1, 5] from start to end of the training. With $\tau = \infty$, the activation function becomes a step function: $f(x) = \tanh(x \cdot \infty) = 1$ if $x > 0$, -1 otherwise. As regularizer on the logits, we use an L2 regularizer for values that are above -1: $\ell_2 = \max(x + 1, 0)^2$. This regularizer gives the interaction variables a bias towards being 0 / -1 for the default, observational case. We add this regularizer with a weight of 5e-4 to the negative log-likelihood loss of BISCUIT.

iVAE As conditioning variable u , we use the concatenation of the latents of the previous time step, z^{t-1} , and the current regime variable R^t . To account for the parameter difference to BISCUIT and the other baselines, we use a 2-layer MLP with hidden size 256. The prior outputs a Gaussian per latent with learnable mean and standard deviation.

LEAP We base our implementation of LEAP (Yao et al., 2022b) on the publicly released code². To allow for LEAP to work both on single images and sequences, we do not use an RNN structure in the encoder as this showed to give poor performance on this dataset. We use the same MLP architecture as in BISCUIT to model the prior network, and use affine conditional normalizing flows for the mapping of the noise to the prior distributions. The discriminator is implemented with a 2-layer MLP with hidden dimension 64. We finetuned the hyperparameters for sparsity and discriminator weight on the NLL weight, for which we found 0.01 and 0.1 respectively.

DMS We base our implementation of DMS (Lachapelle et al., 2022b) on the publicly released code³. In DMS, we use R^t as two action variables that are concatenated to the inputs. We differ from the original implementation by only training the model on the conditional distribution, *i.e.*, $p(X^t | X^{t-1}, R^t)$, and not include the loss for the first element, $p(X^0 | R^0)$. The reason for this is that we observed significantly worse performance on this dataset when including this prior loss, and to be fair with BISCUIT, we align the implementation to focus on the conditional distribution too. As a hyperparameter, we finetuned the sparsity weight in the range [0.001, 0.1] with a final value of 0.02, which lead to a learned graph density of ~ 0.2 .

C.1.3 Results

Result Table As supplement to Figure 6 of the main plot, we provide the R^2 scores with standard deviations in Table 5. Besides the R^2 score, we also report the Spearman correlation on the same latent variables to the ground truth causal variables. We created five datasets with different graphs and mechanisms for each setup. Each model was trained on these five datasets with two different seeds each. This gives us 10 results per model and data setup.

²<https://github.com/weirayao/leap>

³https://github.com/slachapelle/disentanglement_via_mechanism_sparsity

Table 5: Result table for the Voronoi results visualized in Figure 6. All results are reported over 10 experiments, with standard deviation listed next to the mean.

# variables	Interactions	Method	R^2 -diag	R^2 -sep	Spearman-diag	Spearman-sep
6 variables	Robotic arm	BISCUIT	0.99 ± 0.00	0.02 ± 0.03	0.99 ± 0.00	0.07 ± 0.04
		DMSVAE	0.63 ± 0.09	0.31 ± 0.05	0.77 ± 0.07	0.53 ± 0.05
		LEAP	0.62 ± 0.08	0.36 ± 0.06	0.76 ± 0.07	0.57 ± 0.05
		iVAE	0.48 ± 0.06	0.23 ± 0.04	0.64 ± 0.06	0.45 ± 0.04
9 variables	Robotic arm	BISCUIT	0.98 ± 0.00	0.02 ± 0.02	0.99 ± 0.00	0.08 ± 0.04
		DMSVAE	0.67 ± 0.07	0.31 ± 0.03	0.79 ± 0.05	0.53 ± 0.03
		LEAP	0.65 ± 0.06	0.35 ± 0.06	0.78 ± 0.06	0.57 ± 0.05
		iVAE	0.38 ± 0.05	0.23 ± 0.04	0.54 ± 0.06	0.46 ± 0.03
6 variables	Minimal inter.	BISCUIT	0.99 ± 0.00	0.02 ± 0.02	0.99 ± 0.00	0.09 ± 0.03
		DMSVAE	0.62 ± 0.14	0.26 ± 0.05	0.75 ± 0.11	0.48 ± 0.05
		LEAP	0.59 ± 0.07	0.31 ± 0.04	0.74 ± 0.05	0.53 ± 0.04
		iVAE	0.47 ± 0.08	0.25 ± 0.04	0.63 ± 0.07	0.46 ± 0.04
9 variables	Minimal inter.	BISCUIT	0.98 ± 0.00	0.00 ± 0.01	0.99 ± 0.00	0.05 ± 0.02
		DMSVAE	0.62 ± 0.07	0.26 ± 0.03	0.77 ± 0.06	0.48 ± 0.03
		LEAP	0.60 ± 0.07	0.30 ± 0.04	0.74 ± 0.07	0.52 ± 0.04
		iVAE	0.37 ± 0.05	0.22 ± 0.03	0.52 ± 0.07	0.45 ± 0.03

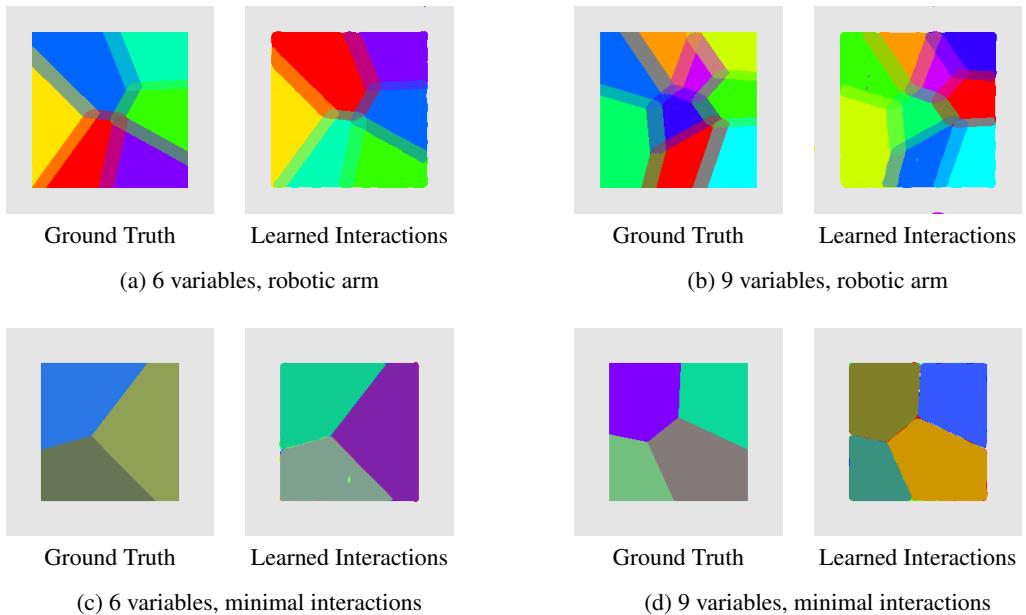


Figure 10: Visualizing the ground truth (left) and learned (right) interaction variables of BISCUIT on the Voronoi dataset. The x and y dimension represent the two dimensions of the robotic arm position $R^t \in [-1.5, 1.5]^2$, and the different colors correspond to interactions with different causal variables, *i.e.*, interaction variables being equal to 1 (not to confuse with the actual colors in the observations). Gray represents the observational regime. (a,b) Under the robotic arm interaction setup, the interaction map corresponds to the same Voronoi structure as the observation. BISCUIT identifies the same structure well up to minor errors on the boundaries. Note that the ground truth and prediction matches up to permutation of the colors, *i.e.*, the same permutation as latent to causal variables. (c,d) Under minimal interactions, BISCUIT models the same interaction structure as the ground truth. The different colors are due to the several overlaps of interaction variables.

Learned Interaction Variables To verify that the learned interaction variables are matching the ground truth, we plot them over the different values of R^t in Figure 10. The x and y dimensions of the images correspond to the two dimensions of R^t , and different colors show different interaction variables. Overall, we find that BISCUIT learned the same underlying structure of the interaction variables, but, as expected, with an arbitrary permutation.

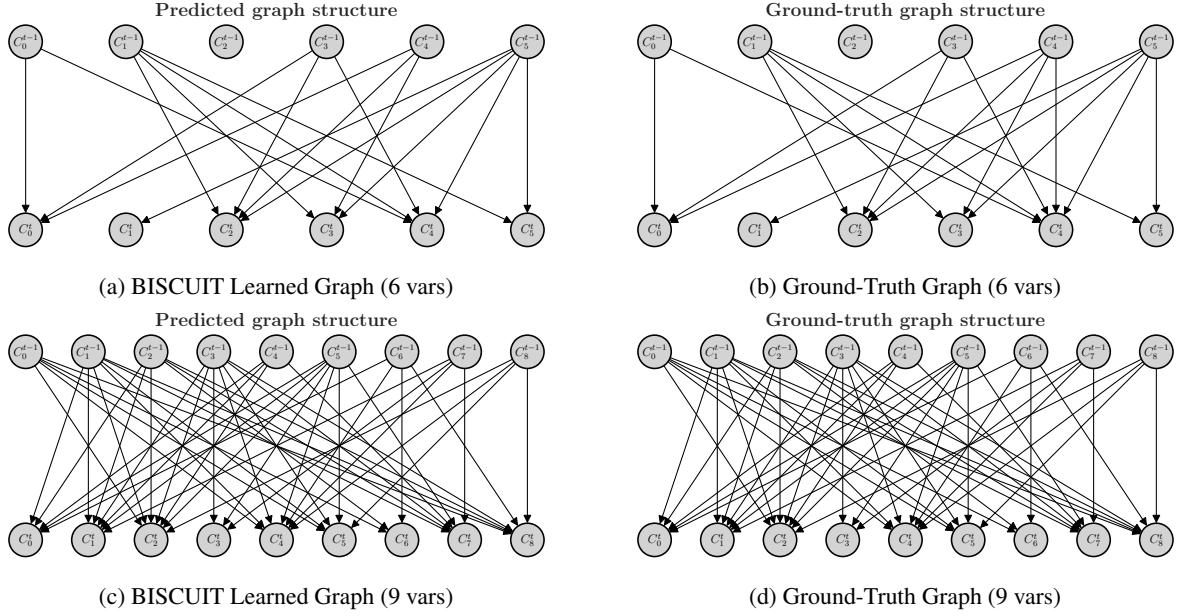


Figure 11: Discovering the causal graph among learned causal variables. Subfigure (a) and (c) show the causal structure that has been discovered between the learned causal variables of BISCUIT on a Voronoi dataset of 6 and 9 variables, respectively. Subfigures (b) and (d) show the corresponding ground truth causal structures. BISCUIT identifies the graph up to an SHD error of 1 for the 6 variables and 0 for 9 variables.

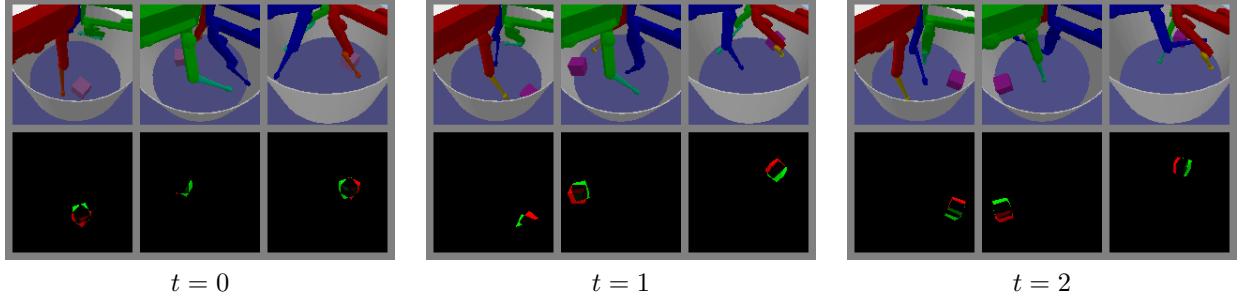


Figure 12: Example sequence in the CausalWorld dataset (Ahmed et al., 2020). Each observation has 3 images from different camera perspectives, with each image being an RGB with additional velocity channel of the object. For visualization, the velocity channel is shown with positive values in red and negative in green. The overall observation dimension is $64 \times 64 \times 12$.

Learned Causal Structure We provide results for estimating the causal graph between the learned causal variables of BISCUIT in the Voronoi dataset. For estimating $p(z^t | z^{t-1}, R^t)$, we start from a fully-connected graph from z^{t-1} to z^t , and applying a sparsity regularizer to remove edges. This is similar to NOTEARS (Zheng et al., 2018) without the acyclicity regularizer, since the directions of all edges are known. Alternatively, intervention-based causal discovery algorithms like DCDI (Brouillard et al., 2020) or ENCO (Lippe et al., 2022a) could be used with the learned interaction variables. The results in Figure 11 show that the identified causal graph matches the ground truth graph, with an SHD of 0 for the 9 variable graph and an SHD of 1 for the 6 variable graph.

C.2 CAUSALWORLD

C.2.1 Dataset Setup

We set up the CausalWorld environment (Ahmed et al., 2020) to contain a single cube as an object with a tri-finger system to interact with it. The environment is observed by three cameras positioned around the arena, each returning an RGB image of dimensions $128 \times 128 \times 3$. Additionally, to perceive the velocity, a difference of frames on the cube is concatenated to

each RGB image. This gives a combined observation size of $128 \times 128 \times 12$. To reduce the computation cost of training all models on this dataset, we bi-linearly downscale the images to a resolution of $64 \times 64 \times 12$.

The regime variable consists of the three rotation angles of each arm of the tri-finger at the current and previous time step, giving overall $R^t \in \mathbb{R}^{18}$. This provides both location and velocity information about the tri-finger robotic system. Furthermore, in this setup, the regime variable across time steps has a causal relation, *i.e.*, $R^{t-1} \rightarrow R^t$, since they share one time step and the tri-fingers have a limited distance they can travel within one time step.

The causal graph consists of seven high-level causal variables: the colors of the three tri-fingers, the friction of the canvas, the floor, and the cube, and finally the state of the cube (cube position, velocity and rotation). The frictions of the different objects are visualized by the colors of the respective objects. For all colors and frictions, we use an additive Gaussian noise model as a ground truth causal model, where under no interactions, we have $C_i^t = 0.95 \cdot C_i^{t-1} + \epsilon_i$, $\epsilon_i \sim \mathcal{N}(0, 0.15)$. Under interactions, we set $C_i^t = \sigma^{-1}(u)$, $u \sim U(0, 1)$ where σ^{-1} is an inverse sigmoid. The causal model of the cube state is based on the physical interactions between the cube, the robot, and the frictions of the floor, stage, and cube. An interaction of one of the tri-fingers with the cube causes an intervention on the color of the touched tri-finger and the cube state (position, velocity, and rotation). Further, the floor friction is randomly re-sampled if all three tri-fingers touch the floor. Similarly, the canvas friction changes if all tri-finger rotations of the first arm element are above a certain threshold. Finally, the robot interacts with the cube friction if the fingers touch in the center of the arena.

We generate 200 sequences of each 1000 frames for training, and 25 sequences for testing. Examples are shown in Figure 12.

C.2.2 Hyperparameters and Implementation Details

BISCUIT We apply the autoencoder + normalizing flow setup of BISCUIT. For this, we first train an autoencoder with mean-squared error loss on individual observations to map the $64 \times 64 \times 12$ input to a 32 dimensional latent space. During the autoencoder training, we apply a Gaussian noise of 0.05 on the latents, and add an L2 regularizer with a weight of 1e-5 to limit the scale of the latent variables. As architecture, we use a convolutional ResNet (He et al., 2016) with two convolutional layers with consecutive GroupNorm normalization (Wu et al., 2018) per ResNet block. After each two ResNet blocks, we reduce the spatial dimensionality using a convolution with stride 2 until we reach a spatial size of 4×4 . At this point, the feature map is flattened, and two linear layers map it to the 32 latent dimensions. The decoder is a mirrored version of the encoder, replacing stride convolutions with up-scaling layers using bi-linear interpolation. We add R^t to the decoder by concatenating it with the latent vector before passing it to the first linear layer of the decoder. Both encoder and decoder use a channel size of 128. We train this network with a batch size of 128 and learning rate of 4e-4 with cosine scheduling for 500 epochs.

The normalizing flow follows the architecture used by Lippe et al. (2022b), namely six autoregressive affine coupling layers (Dinh et al., 2017) with 1x1 convolutions and activation normalization (Kingma et al., 2018) in between. The prior uses the same setup and hyperparameters as for the Voronoi dataset, except using a hidden dimension of 64 instead of 32 in the MLPs. We use a batch size of 512 and learning rate of 1e-3, and train for 100 epochs.

Baselines For all baselines, we use the same convolutional ResNet architecture for the encoder and decoder as in the autoencoder of BISCUIT. To account for the additional parameters introduced by the normalizing flow of BISCUIT, we increase the hidden dimensions of the prior networks correspondingly. However, in general, we found no noticeable gain from increasing the prior dimensions of the baselines further beyond 64 for LEAP and DMS per latent, and 51sarta2 for iVAE. We performed a small hyperparameter search over the learning rates {2e-4, 4e-4, 1e-3} and sparsity regularizers {0.001, 0.01, 0.1}. For LEAP and iVAE, we picked learning 4e-4, and 2e-4 for DMS. For the sparsity regularizers, very high regularizer lead to early sparsification of the graph, while too small regularizers did not change the graph much. Thus, we picked 0.01 for both models, although it did not show much of an impact on the final identification result. We use a batch size of 64 and train for 250 epochs with cosine learning rate scheduling. Longer trainings did not show any improvements.

C.2.3 Results

Correlation Evaluation We show the results of all methods on the dataset in Table 6. Similar to the results on the Voronoi dataset, we include the standard deviation over multiple seeds. We performed each experiment only for three seeds to limit the computational cost, and the standard deviation was much lower than the significant differences. As a second metric, we also report the Spearman correlation, which shows a very similar trend. Finally, we show example R^2 matrices learned by all methods in Figure 13.

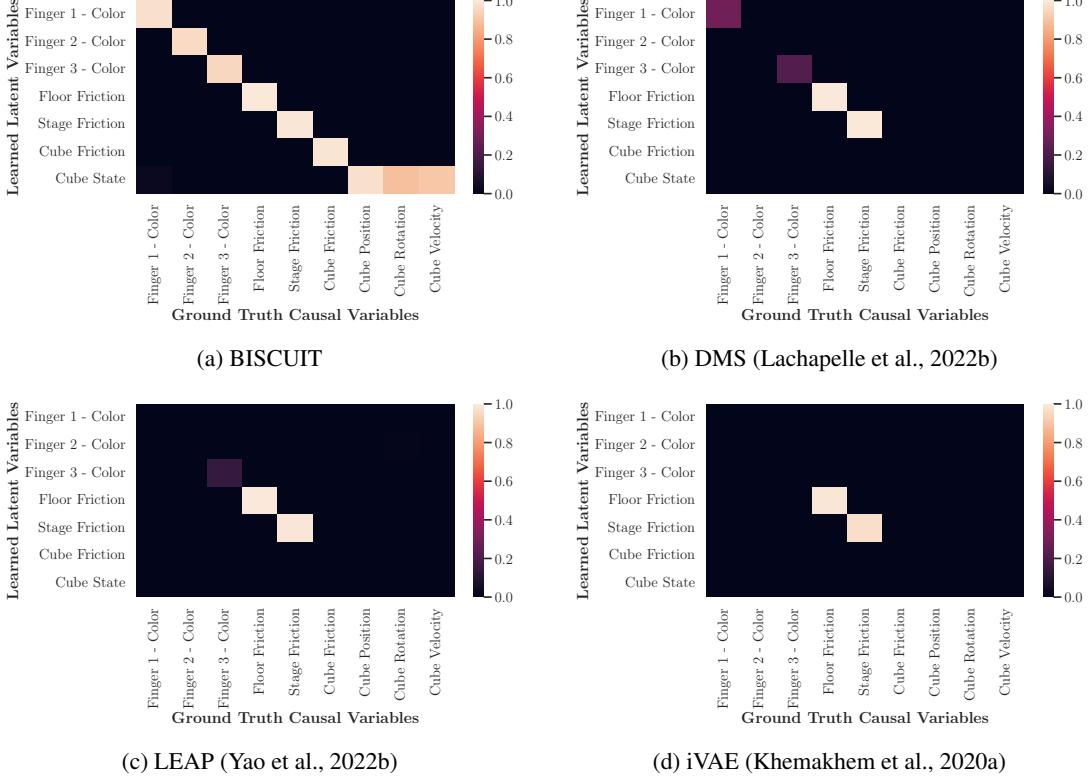


Figure 13: R^2 matrix for learned models on the CausalWorld environment (Ahmed et al., 2020): (a) BISCUIT (ours), (b) DMS (Lachapelle et al., 2022b), (c) LEAP (Yao et al., 2022b), (d) iVAE (Khemakhem et al., 2020a).

Table 6: Result table for the CausalWorld experiments over 3 seeds, with standard deviation listed next to the mean.

Method	R^2 -diag	R^2 -sep	Spearman-diag	Spearman-sep
BISCUIT	0.97 ± 0.01	0.01 ± 0.00	0.98 ± 0.00	0.09 ± 0.02
DMSVAE	0.32 ± 0.03	0.00 ± 0.00	0.44 ± 0.09	0.02 ± 0.00
LEAP	0.30 ± 0.02	0.00 ± 0.00	0.36 ± 0.06	0.02 ± 0.00
iVAE	0.28 ± 0.00	0.00 ± 0.00	0.29 ± 0.00	0.02 ± 0.00

Reconstructions The identification of the baselines suffers from the poor reconstruction of the models. We show an example of the reconstructions in Figure 14. In general, all baselines miss the cube as well as the colors of the tips. This is because the robotic arms and the cube move over time and appear at different positions in different frames, requiring the VAE to be accurately modeling their positions before learning the colors. However, the gradient signal is often too small to overcome the KL regularizer on the latent space. Common tricks like using a KL scheduler did not show to improve the results. In comparison, BISCUIT-NF can accurately reconstruct the images in Figure 15. Since it uses an autoencoder with an unregularized latent space, it is much easier for the model to map all information in the latent space.

Interaction Variables To analyze the learned interaction variables in BISCUIT, we recorded in the simulator the time steps on the test set in which there is a collision between an arm and the cube, and convert it to a binary signal (0 - no collision, 1 - collision). After training BISCUIT, we compare the learned binary interaction variables to the recorded collisions in terms of F1 score, similar to the Voronoi experiments. We follow a similar procedure for the remaining causal variables as well. We plot the F1 matrix of learned vs ground truth interactions in Figure 16. The learned interaction variables for each arm have an F1 score of about 50%. Since collisions only happen in approx. 5% of the frames, a score of 50% indicates a high similarity between the learned interaction and the ground truth collisions. The mismatches are mostly due to the learned interaction being more conservative, i.e. being 1 already a frame too early sometimes.

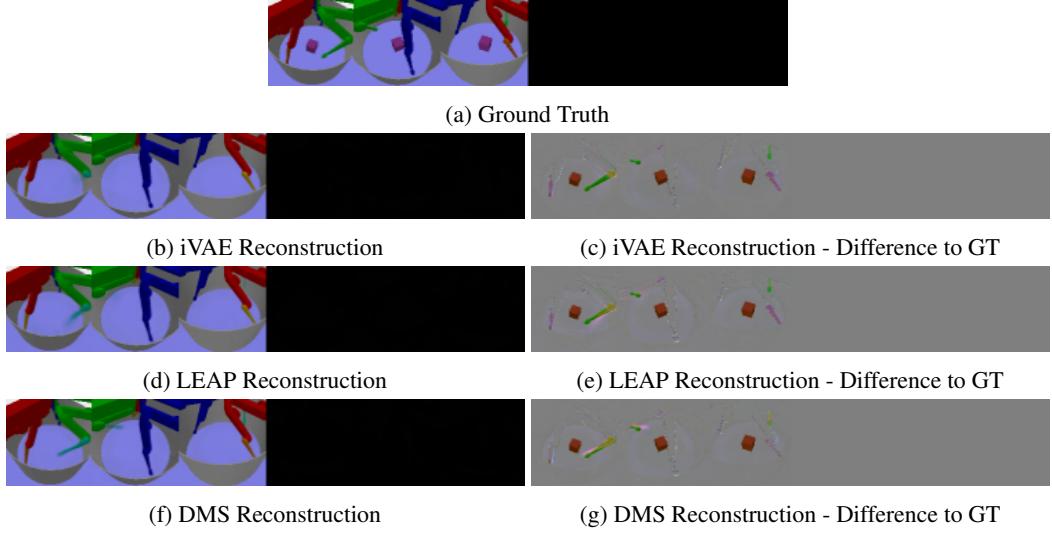


Figure 14: Reconstructions on the CausalWorld dataset by the baselines (b) iVAE, (d) LEAP and (f) DMS for an (a) ground truth example. The differences between the prediction of iVAE, LEAP and DMS to the ground truth is shown in (c), (e), (f), respectively. For better visual representation, we scale the difference from $[-1, 1]$ back to $[0, 1]$, *i.e.*, no difference being gray. All baselines struggle to reconstruct the robotic arms, in particular their color, and the cube.

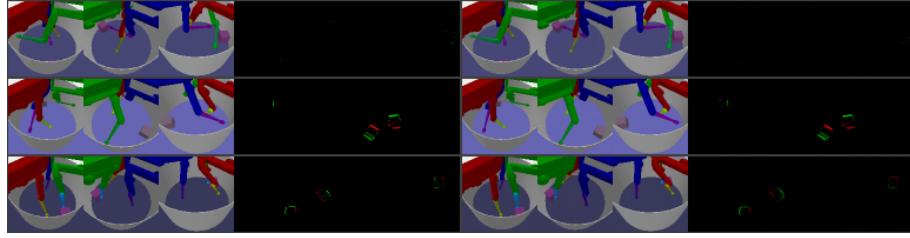


Figure 15: Reconstructions of BISCUIT on the CausalWorld dataset. **Left** is shown the ground truth observation of the environment, and **right** the reconstruction of the autoencoder in BISCUIT-NF. BISCUIT can reconstruct the images accurately up to minor smoothing artifacts.

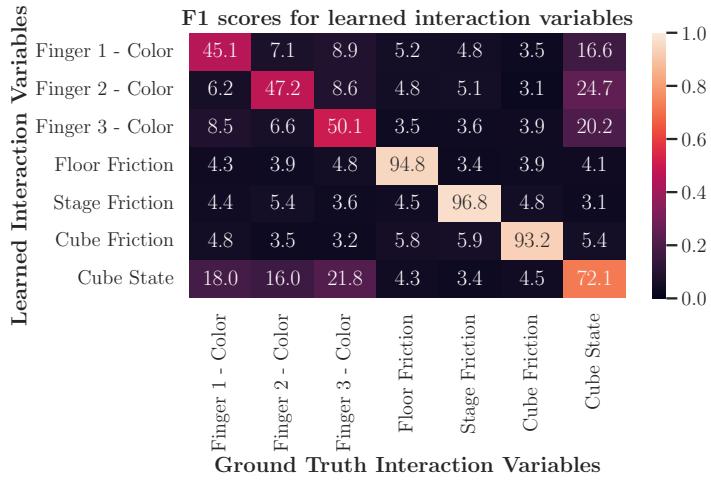


Figure 16: F1 scores between the learned interaction variables by BISCUIT and ground truth interactions on the CausalWorld dataset. BISCUIT closely models the interactions between the causal variables.

Frames				
Performed action	-	PickupObject - Plate	CloseObject - Microw.	PutObject - Counter
Time step	$t = 0$	$t = 1$	$t = 2$	$t = 3$

Figure 17: Example sequence in the iTHOR dataset (Kolve et al., 2017). At each time step, we perform one action in the environment and use a randomly sampled pixel location of the interacted object as regime variable. The resolution of each frame is $256 \times 256 \times 3$.

Table 7: List of the objects in the iTHOR environment with their corresponding possible actions to performed on.

Objects	Possible actions
Plate	PickupObject, PutObject, MoveObject
Egg	PickupObject, PutObject, MoveObject
Cabinet	OpenObject, CloseObject
Microwave	OpenObject, CloseObject, ToggleObject
Toaster	ToggleObject
Stove-Knob (4×)	ToggleObject

C.3 ITHOR

C.3.1 Dataset Setup

The iTHOR dataset (Kolve et al., 2017) is based on the `FloorPlan10` environment, which is the default kitchen environment. We show the overall floor plan in Figure 18. In it, we position the robot in front of the kitchen counter, and keep its position fixed. As a first step in the environment, we place two movable objects (a plate with a potato and an egg) randomly on the counter, as well as the pan on the stove. We remove all remaining movable objects from the robot’s view. Then, at each time step, we perform a randomly chosen action on one of the objects. An overview of all objects and actions is shown in Table 7. Note that not all actions are always possible. For instance, objects can only be opened when they are closed, and vice versa. Further, we can perform the action `ToggleObject` on the microwave only if the microwave is closed, and the action `OpenObject` when the microwave is turned off. For the movable objects, we can only pick up one of the two objects at once. When an object is picked up, we can interact with a remaining object on the counter with the action `MoveObject` which moves it to a new random position on the counter. The action `PutObject` randomly chooses one of the available receptacles to put the object down on. For the plate, this includes the counter and the microwave if it is open. For the egg, this includes the counter and the pan. If the egg is put into the pan, it is automatically broken and cannot be picked up or moved anymore. When any object is put down to the counter, we randomly sample a location on the counter which does not overlap with any other object on the counter. Besides these interactions, we add the `NoOp` action, which does not interact with any object and represents the observational regime.

The regime variable $R^t \in [0, 1]^2$ represents the click-location of a user on the image to select which object to interact with. Specifically, after choosing the action-object pair to perform at a time step, we use the object segmentation mask of the iTHOR simulator to identify the set of pixels that show the object in question. We then randomly sample one of these pixels, and set R^t as the location of this pixel in the frame. After that, the actual action is performed. As location for the `NoOp` action, we sample a pixel location which does not belong to any object in the current frame. For the microwave, we split the object in two halves, where the left part, *i.e.*, the door, performs the action `OpenObject`, and the right part, *i.e.*, the buttons



Figure 18: The floor plan of the `FloorPlan10` environment (Kolve et al., 2017). The robot’s position and orientation are shown in red.

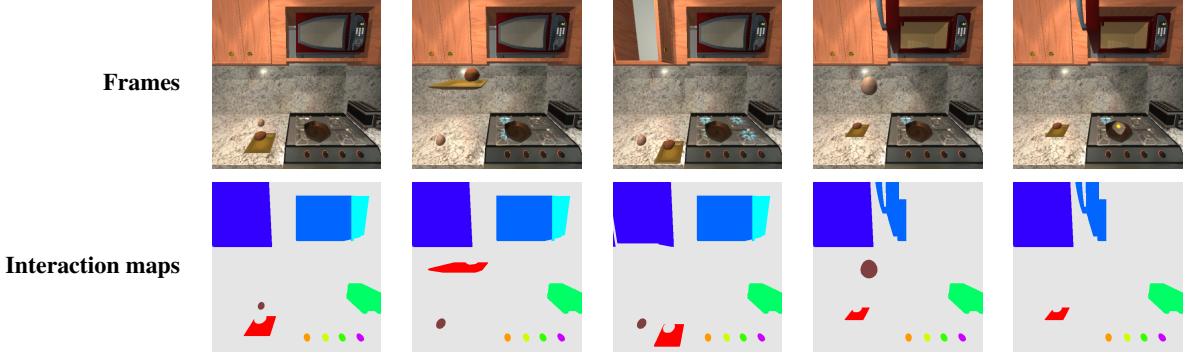


Figure 19: Example ground truth interaction maps in the iTHOR dataset (Kolve et al., 2017). The colors are aligned with the predicted interaction maps of BISCUIT in Figure 7.

and display, activates the microwave. Additionally, when the microwave is open, we sample the pixel location from the open door. Finally, the stoves are controlled by the knobs, such that their interactions are sampled from the knobs positions. Examples of the interaction maps are shown in Figure 19.

The causal variables in this environment align with the action-object pairs at hand. For each object that has the action `ToggleObject`, we have a binary causal variable indicating whether it is active or not. Similarly, for each object that has the action `OpenObject`, we have a binary causal variable indicating whether it is open or not. For each object that has the action `PickupObject`, we have one binary causal variable indicating whether it is picked up or not, and three causal variables for the object’s x-y-z position in the 3D environment. Additionally, for the egg, we have two binary causal variables indicating whether it is broken, *i.e.*, in the pan, and cooked, which happens instantaneously when the stove is being turned on and the egg is in the pan. This results in overall 18 causal variables:

- Cabinet-Open
- Egg-Broken, Egg-Pos-x, Egg-Pos-y, Egg-Pos-z, Egg-Cooked, Egg-PickedUp
- Microwave-Open, Microwave-Active
- Plate-Pos-x, Plate-Pos-y, Plate-Pos-z, Plate-PickedUp
- Stove1-Active, Stove2-Active, Stove3-Active, Stove4-Active
- Toaster-Active

We generate the frames with a resolution of $512 \times 512 \times 3$, and reduce the dimension to $256 \times 256 \times 3$ via bi-linearly interpolation afterward. The high resolution is required since some states are only shown by fine details in the image. For instance, Figure 20 shows that the difference between the toaster being active or not is only a few pixels. In the meantime, actions like the cabinet being opened change about 10% of the image. This makes it challenging for reconstruction-based methods to fully capture all variables fairly in the latent space. Besides, the environment uses a 3D physics engine, showing some interactions that we are not able to fully capture by the causal variables defined. For instance, when the stove is turned on, the flame slowly grows over time and slightly fluctuates once it reached its maximum. Further, when the egg is broken, it slides into the pan over the next three frames.

We generate 1500 sequences of each 100 frames, and 250 sequences for testing. Examples are shown in Figure 17.

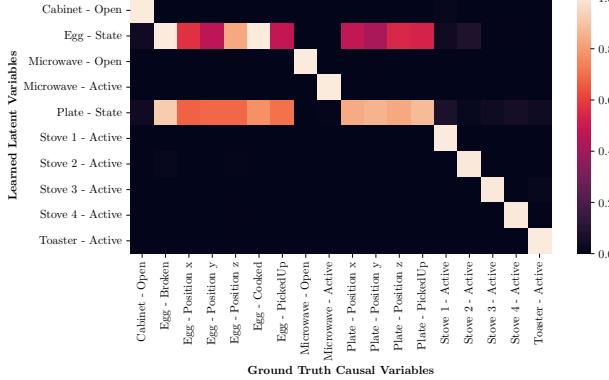
C.3.2 Hyperparameters and Implementation Details

BISCUIT We slightly adapt the setup from the experiments of the CausalWorld environment by:

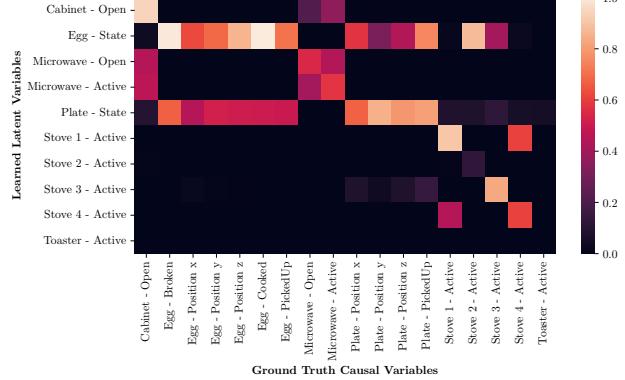
1. we reduce the channel size to 64 and batch size to 64 due to the higher resolution;
2. we increase the latent dimension to 40 due to the larger number of causal variables;
3. we reduce the number of epochs to 150 for the autoencoder;
4. we reduce the learning rate to 2e-4;
5. we do not use R^t as input to the decoder.



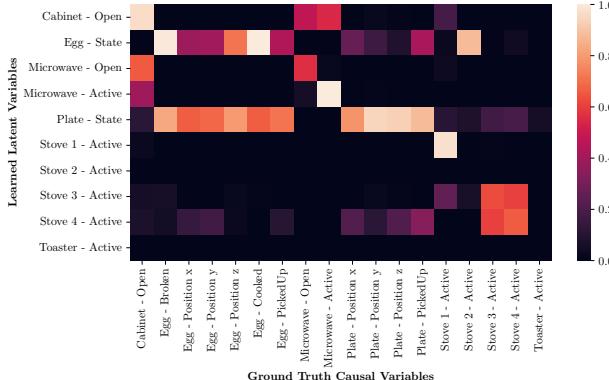
Figure 20: The Toaster being turned off or on changes very few pixels in the observation.



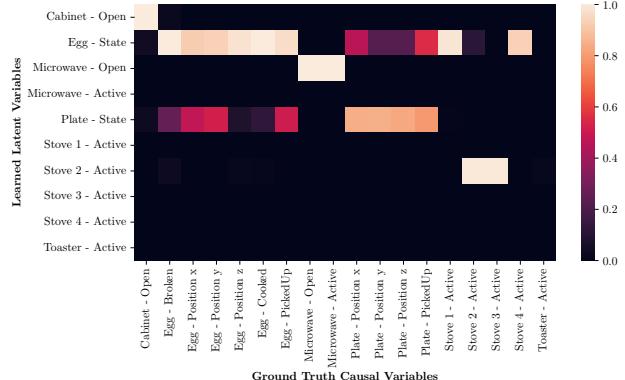
(a) BISCUIT



(b) DMS (Lachapelle et al., 2022b)



(c) LEAP (Yao et al., 2022b)



(d) iVAE (Khemakhem et al., 2020a)

Figure 21: R^2 matrix for learned models on the iTHOR environment (Kolve et al., 2017): (a) BISCUIT (ours), (b) DMS (Lachapelle et al., 2022b), (c) LEAP (Yao et al., 2022b), (d) iVAE (Khemakhem et al., 2020a).

Table 8: Result table for the iTHOR experiments over 3 seeds, with standard deviation listed next to the mean.

Method	R^2 -diag	R^2 -sep	Spearman-diag	Spearman-sep
BISCUIT	0.96 ± 0.01	0.15 ± 0.01	0.96 ± 0.00	0.22 ± 0.02
DMSVAE	0.61 ± 0.05	0.40 ± 0.04	0.65 ± 0.07	0.48 ± 0.05
LEAP	0.63 ± 0.04	0.45 ± 0.04	0.65 ± 0.06	0.49 ± 0.03
iVAE	0.48 ± 0.04	0.35 ± 0.02	0.48 ± 0.04	0.38 ± 0.02

In general, we did not find BISCUIT to be particularly hyperparameter sensitive in these experiments.

Baselines We adapt the experimental setup of the baselines in the same way as BISCUIT. We set the learning rate to 2e-4 for all baselines, which was more stable across models.

C.3.3 Results

Correlation Evaluation Table 8 supplements the results of Table 1 in the main text by including standard deviations over three seeds, as well as the Spearman scores on the same variables. For the multidimensional causal variables ‘Egg state’ and ‘Plate state’, where each dimension is highly correlated and shares the same interactions, we follow Lippe et al. (2022b) by averaging the variable’s dimensions in the R^2 -diag calculation before taken the diagonal average. An example R^2 matrix for each method is shown Table 8. BISCUIT identifies and separates all causal variables up to the two movable objects (plate and egg). Meanwhile, for both DMS and LEAP, we find that the models diversely entangle the causal variables. The only

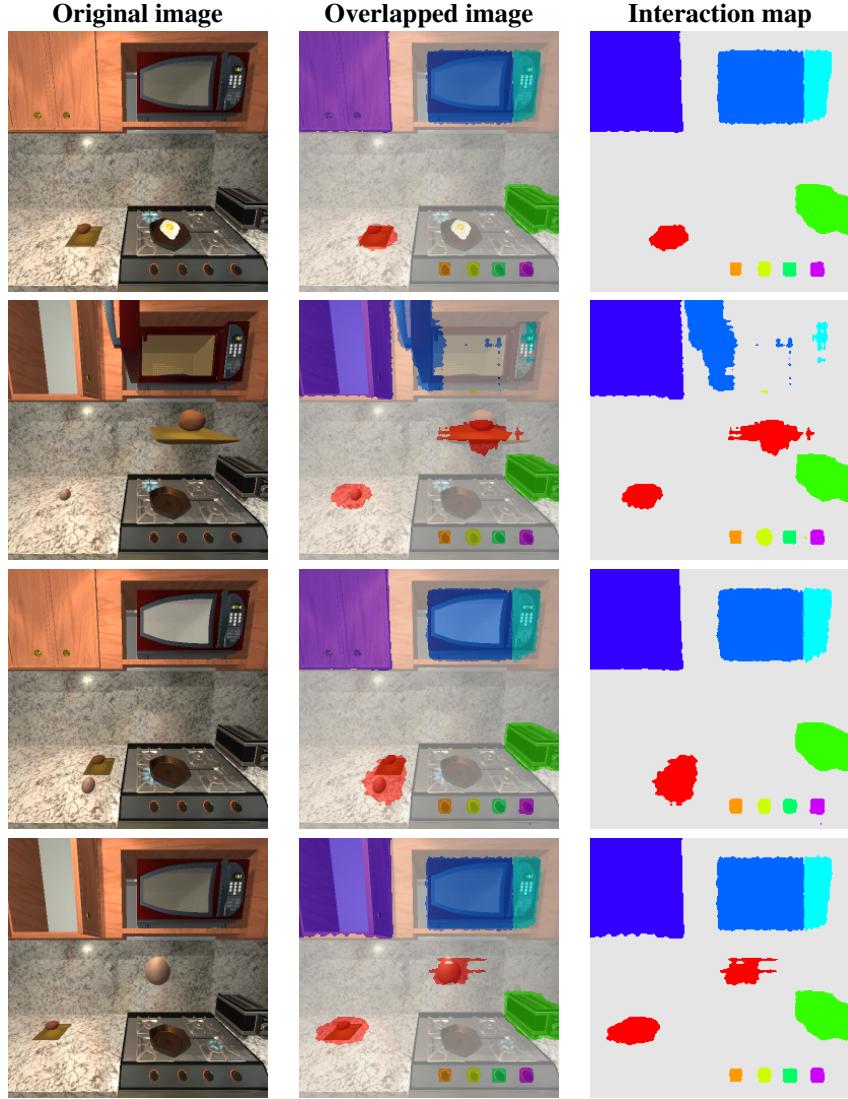


Figure 22: Learned interaction maps of BISCUIT on iTHOR (Kolve et al., 2017). The images on the left constitute the input image, the right image the learned interaction variables, and the center the overlap of both. The colors correspond to the following causal variables modeled by the respective latent variable: **red** → plate/egg state, **light green** → Toaster-Active, **light blue** → Microwave-Active, **blue** → Microwave-Open, **dark blue** → Cabinet-Open, **orange** → Stove1-Active, **yellow** → Stove2-Active, **green** → Stove3-Active, **purple** → Stove4-Active.

causal variable that has not been modeled at all is the toaster, which is likely due to its small pixel footprint in the image (see Figure 20). Interestingly, iVAE showed a very different trend. While it has less diverse entanglement than LEAP and DMS, it often models several binary causal variables in the same latent dimension. Since its latent variables are all conditioned on full R^t , there is no difference to model two binary variable in one or two separate dimensions. In contrast, this does not happen in BISCUIT since these different binary causal variables have different binary interaction variables.

Interaction Maps Similarly to Figure 7 in the main text, we visualize more examples of the learned interaction maps of BISCUIT in Figure 22. The first column shows the input image, the third the learned binary interaction variables, and the second column the overlap of both. One can clearly see how BISCUIT adapts its interaction variables to the input image. For instance, in the second row, one can see how BISCUIT identifies the regions of the plate and the egg, visualized in red. In line with the R^2 evaluation, BISCUIT maps the interactions with the plate and egg together into one interaction variable instead of separating them. The interaction variables of the microwave change towards focusing on the opened door for the action `CloseObject` (blue). Since it is not possible to activate the microwave at this state, its corresponding interaction

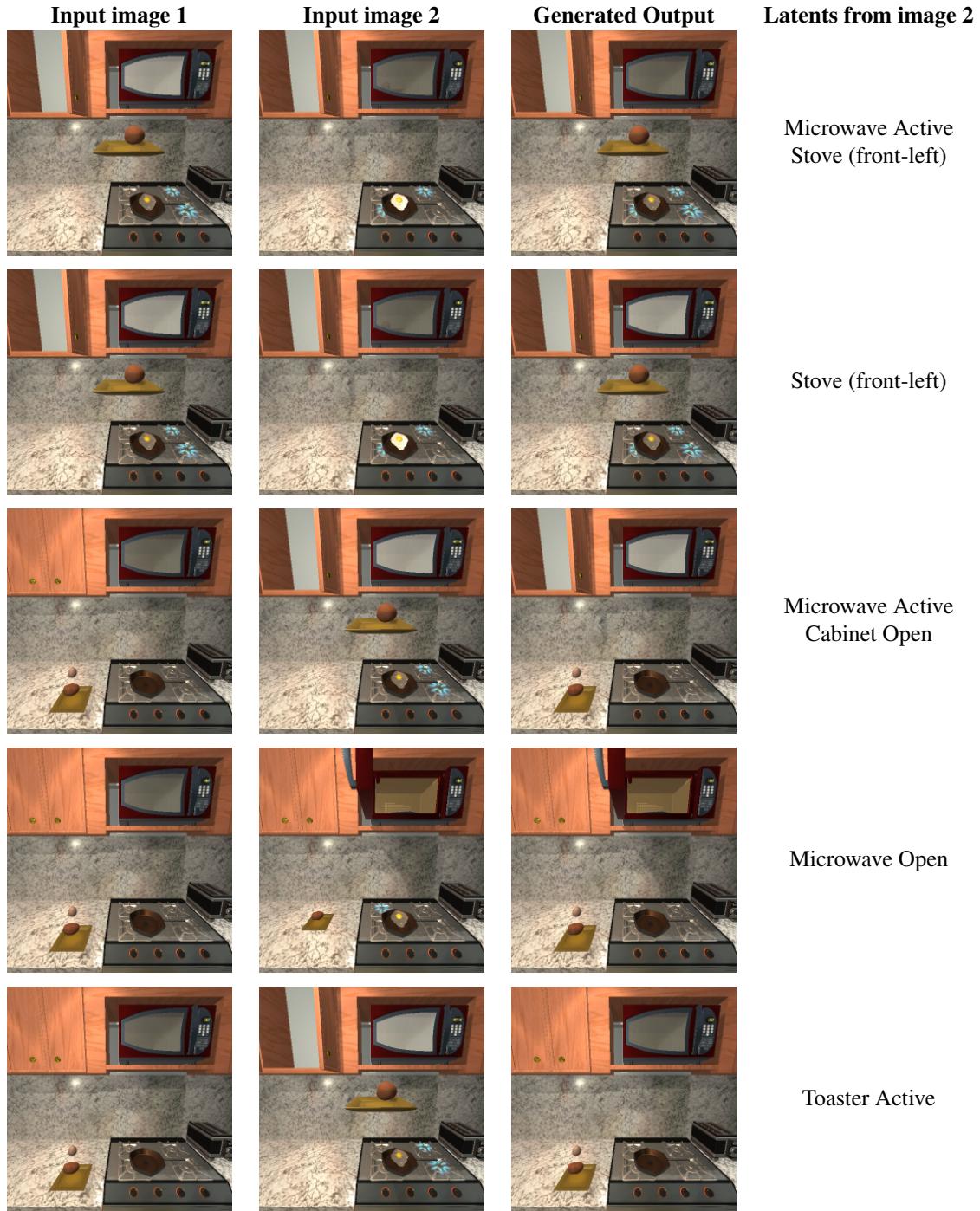


Figure 23: Additional triplet predictions of BISCUIT on images from the test dataset. The first two images are encoded in the latent space. We then replace the latents of the first image by the second image for the dimensions that correspond to the causal variable listed in the fourth column. The third column shows then the reconstruction of BISCUIT for this new latent vector.

variable (light blue) mostly disappears. It does not fully disappear, since the model has never seen a click in this region when the microwave door is open. Meanwhile, other interaction variables that are independent of the actual input image, *e.g.*, the stove knobs or the toaster, are also kept constant throughout the examples by BISCUIT.

The images only show 9 out of the 40 learned interaction maps. The remaining 31 interaction variables are mostly either (1) constants and would give an empty interaction map, or (2) follow a very similar pattern to the red interaction variable, *i.e.*, the plate and the egg. While we find that some of these dimension focus slightly more on the plate and some more on the

egg, there was none that fully focused on either variable, which follows the insights of the R^2 evaluation.

Triplet Generations Similar to Figure 8 in the main text, we visualize additional examples of triplet predictions of BISCUIT in Figure 23. The first two images represent the input images, and the third the generated output. The fourth column specifies which causal variables we tried to replace in the first image by the latents of the second image. BISCUIT consistently generates the correct counterfactual prediction for a variety of causal variables and input images.