Disentanglement of Latent Representations via Causal Interventions

Gaël Gendron , **Michael Witbrock** , **Gillian Dobbie**

University of Auckland

ggen187@aucklanduni.ac.nz, m.witbrock@auckland.ac.nz, g.dobbie@auckland.ac.nz

Abstract

The process of generating data such as images is controlled by independent and unknown factors of variation. The retrieval of these variables has been studied extensively in the disentanglement, causal representation learning, and independent component analysis fields. Recently, approaches merging these domains together have shown great success. Instead of directly representing the factors of variation, the problem of disentanglement can be seen as finding the interventions on one image that yield a change to a single factor. Following this assumption, we introduce a new method for disentanglement inspired by causal dynamics that combines causality theory with vector-quantized variational autoencoders. Our model considers the quantized vectors as causal variables and links them in a causal graph. It performs causal interventions on the graph and generates atomic transitions affecting a unique factor of variation in the image. We also introduce a new task of action retrieval that consists of finding the action responsible for the transition between two images. We test our method on standard synthetic and real-world disentanglement datasets. We show that it can effectively disentangle the factors of variation and perform precise interventions on high-level semantic attributes of an image without affecting its quality, even with imbalanced data distributions.

1 Introduction

The problem of recovering the mechanisms underlying data generation, particularly for images, is challenging and has been widely studied in machine learning research [Locatello *et al.*, 2019; Gresele *et al.*, 2021; Schölkopf *et al.*, 2021; Lachapelle *et al.*, 2022]. The disentanglement field aims to represent images as high-level latent representations where such mechanisms, or *factors of variation*, are divided into separate, e.g. orthogonal, dimensions [Higgins *et al.*, 2018]. By contrast, causal representation learning attempts to recover such factors as causal variables sparsely linked in a graph [Schölkopf *et al.*, 2021; Ahuja *et al.*, 2022]. Despite the similarities between the two problems, until recently little

work has attempted to combine the two fields [Suter *et al.*, 2019; Yang *et al.*, 2021]. Some approaches have also borrowed ideas from independent component analysis [Gresele *et al.*, 2021; Lachapelle *et al.*, 2022]. A central concept linking this work is the Independent Causal Mechanisms (ICM) principle [Schölkopf *et al.*, 2021] which states that the generative process of a data distribution is made of independent and autonomous modules. In order to recover these modules, disentanglement approaches mainly rely on variational inference and Variational Auto-Encoders (VAEs) [Locatello *et al.*, 2019] or Generative Adversarial Networks [Gabbay *et al.*, 2021]. Despite the success of vector-quantized VAE architectures for generating high-quality images at scale [van den Oord *et al.*, 2017; Razavi *et al.*, 2019; Ramesh *et al.*, 2021], they have not been considered in the disentanglement literature, except in the speech synthesis domain [Williams *et al.*, 2021].

In this paper, we attempt to bridge this gap by proposing a novel way to represent the factors of variation in an image using quantization. We introduce a Causal Transition (CT) layer able to represent the latent codes generated by a quantized architecture within a causal graph and allowing causal interventions on the graph. We consider the problem of disentanglement as equivalent to recovering the atomic transitions between two images X and Y. In this setting, one high-level action causes an intervention on the latent space, which generates an atomic transition. This transition affects only one factor of variation. We use our architecture for two tasks: given an image, act on one factor of variation and generate the intervened-on output; and given an input-output pair, recover the factor of variation whose modification accounts for the difference. To study the level of disentanglement of latent quantized vectors, we also introduce a Multi-Codebook Quantized VAE (MCQ-VAE) architecture, dividing the VQ-VAE latent codes into several vocabularies. Figure 2 illustrates our full architecture. We show that our model can effectively disentangle the factors of variation in an image and allow precise interventions on a single factor without affecting the quality of the image, even when the distribution of the factors is imbalanced.

We summarise our contributions as follows: (i) We introduce a novel quantized variational autoencoder architecture and a causal transition layer. (ii) We develop a method to perform atomic interventions on a single factor of variation

in an image and disentangle a quantized latent space, even with imbalanced data. (iii) Our model can learn the causal structure linking changes on a high-level global semantic concept to low-level local dependencies. (iv) We propose a new task of recovering the action that caused the transition from an input to an output image. (v) We show that our model can generate images with and without interventions without affecting quality. Our code and data are available here: https://github.com/Strong-AI-Lab/ct-vae.

2 Related Work

Disentanglement There is no formal definition of disentanglement, but it is commonly described as the problem of extracting the *factors of variation* responsible for data generation [Locatello *et al.*, 2019]. These factors of variation are usually considered independent variables associated with a semantic meaning [Mathieu *et al.*, 2019; Schölkopf *et al.*, 2021]. Formally, it amounts to finding, for an image X, the factors $Z = \{Z_i\}_{i \in 1...D}$ s.t. $f(Z_1, \ldots, Z_D) = X$, D being the dimensionality of the latent space. Modifying the value of one Z_i modifies a single semantic property of X (e.g. the shape, the lighting, or the pose) without affecting the other properties associated with the values $Z_{j \neq i}$. The main disentanglement methods are based on the regularisation of Variational Autoencoders (VAEs) [Kingma and Welling, 2014]. Unsupervised models comprise the β-VAE [Higgins *et al.*, 2017], the β-TCVAE [Chen *et al.*, 2018], the Factor-VAE [Kim and Mnih, 2018] or the DIP-VAE [Kumar *et al.*, 2018]. However, unsupervised approaches have been challenged, and the claim that fully unsupervised disentanglement is achievable remains under debate [Locatello *et al.*, 2019; Horan *et al.*, 2021]. More recent approaches rely on weak supervision [Locatello *et al.*, 2020; Gabbay *et al.*, 2021]. Our approach belongs to this category. In particular, the Causal-VAE [Yang *et al.*, 2021] generates a causal graph to link the factors of variation together. Our approach also attempts to take advantage of causal models for disentanglement, but the two methods differ greatly. We consider the problem of disentanglement from the perspective of causal dynamics and use quantization instead of a standard VAE to generate the causal variables.

Quantization The Vector-Quantized VAE (VQ-VAE) [van den Oord *et al.*, 2017] is an autoencoder where the encoder generates a discrete latent vector instead of a continuous vector $Z \in \mathbb{R}^D$. From an input image, the VQ-VAE builds a discrete latent space $\mathbb{R}^{K \times D}$ with K vectors representing the quantization of the space. As an analogy, these vectors can be interpreted as words in a codebook of size K. The encoder samples $N \times N$ vectors from the latent space when building $Z \in \mathbb{R}^{N \times N \times D}$. Each sampled vector describes the local information in a $N \times N$ grid representing an abstraction of the input image X. The VQ-VAE and its derivations have proven very successful at generating high-quality images at scale [Razavi *et al.*, 2019; Ramesh *et al.*, 2021]. The Discrete Key-Value Bottleneck [Träuble *et al.*, 2022] builds upon the VQ-VAE architecture, introducing a key-value mechanism to retrieve quantized vectors and using multiple codebooks instead of a single

one; the method is applied to domain-transfer tasks. To the best of our knowledge, we are the first to apply quantized autoencoders to disentanglement problems.

End-to-end Causal Inference Causal tasks can be divided into two categories: causal structure discovery and causal inference. Causal structure discovery consists in learning the causal relationships between a set of variables with a Direct Acyclic Graph (DAG) structure, while causal inference aims to estimate the values of the variables [Pearl, 2009] quantitatively. In our work, we attempt to recover the causal structure responsible for the transition from an input image X to an input image Y and perform causal inference on it to retrieve the values of the missing variables. As the causal graph acts on latent variables, we also need to retrieve the causal variables, i.e. the disentangled factors of variation, Z.

The Structural Causal Model (SCM) [Pearl, 2009] is a DAG structure representing causal relationships on which causal inference can be performed. Causal queries are divided into three layers in Pearl's Causal Hierarchy (PCH) [Bareinboim *et al.*, 2022]: associational, interventional and counterfactual. Our work attempts to solve interventional queries, i.e. questions of the type "how would Y evolve if we modify the value of X?", represented by the formula $P(Y = y|\mathbf{do}(X = x))$. The **do**-operation [Pearl, 2009] corresponds to the attribution of the value x to the variable X regardless of its distribution. The Causal Hierarchy Theorem (CHT) [Bareinboim *et al.*, 2022] states that interventional data is necessary to solve interventional queries. Accordingly, the data we use is obtained by performing interventions a on images X.

Recent work performed causal structure discovery and inference in an end-to-end fashion, like VCN [Annadani *et al.*, 2021] and DECI [Geffner *et al.*, 2022]. Our approach is similar, as we want to identify and estimate the causal links end-to-end. The main differences are that we do not assume linear relationships as in VCN, and the causal variables are unknown in our problem and must also be estimated. This problem of retrieving causal variables is known as causal representation learning [Schölkopf *et al.*, 2021]. In particular, our method is close to interventional causal representation learning [Ahuja *et al.*, 2022].

Graph Neural Networks Graph Neural Networks are a family of Deep Learning architectures operating on graph structures. A graph $\mathcal{G} = \langle V, E \rangle$ is a set of nodes V and edges E where an edge $e_{ij} \in E$ links two nodes v_i and v_j. A feature vector $x_i \in \mathcal{D}$ is associated with each node v_i. A feature matrix $X \in \mathbb{R}^{|V| \times D}$ represents the set of vectors. The graph is represented with an adjacency matrix $A \in [0, 1]^{|V| \times |V|}$. Graph neural networks aggregate the node features based on the neighbourhood of each node. A generic representation is shown in Equation 1.

$$X^{(l+1)} = GNN(X^{(l)}; A) \qquad (1)$$

The most popular GNN architectures are GCN [Kipf and Welling, 2017], GAT [Velickovic *et al.*, 2018], and Graph-SAGE [Hamilton *et al.*, 2017]. Recently, Graph Neural Networks have proved themselves a suitable architecture for causal inference tasks [Zecevic *et al.*, 2021] because of their

ability to represent interventional queries on a causal graph. iVGAE [Zecevic *et al.*, 2021] and VACA [Sánchez-Martín *et al.*, 2022] are variational autoencoders operating on graph structures and able to perform **do**-operations. Our work differs from these models as they assume the causal graph structure to be known, whereas it is unknown in our problem.

Causal Dynamics World models attempt to learn a latent representation capturing the dynamics of an environment over time. The Variational Causal Dynamics (VCD) model [Lei *et al.*, 2022] learns the invariant causal mechanisms responsible for the evolution of an environment under the influence of an action a. In such systems, the environment transits from the state (s) to $(s + 1)$ because of a. We approach disentanglement from this angle, considering our representation disentangled if, when applying an action a_i, we intervene on a single factor of variation Z_i. The state (s) corresponds to the input image X, and the state $(s+1)$ corresponds to an output image Y after intervention a on one factor of variation of X. The main difference between our work and VCD is that our environment is latent and must be discovered.

3 Causal Transition Variational Autoencoder

3.1 Problem Definition

The problem we aim to solve can be divided into two parts. The first is a disentanglement problem; we aim to generate a disentangled latent representation to which we can apply an intervention on a specific feature, e.g. change the pose, the lighting or the shape of the object shown. The second problem is reverse-engineering the intervention responsible for the transition from an input to an output image: given the input image and the output image after an intervention, identify the action that caused the transition.

We name the input image X and the output image after transition Y, with a being the cause of the change. Given a set $\mathcal{S}$ of pairs of input-output images $(X_1, Y_1), (X_2, Y_2), \cdots \in \mathcal{S}$ s.t. $\forall (X, Y) \in \mathcal{S}, Y = f_a(X)$, we aim to find the function f_a. The first problem is returning Y given X and a, and the second problem is extracting a given X and Y. The causal queries associated are $P(Y|X, do(a))$ and $P(a|X, Y)$.

3.2 Overview of Our Method

We generate disentangled latent representations $\mathbf{L_x}$ and $\mathbf{L_y}$, and apply a causal transition model on them to represent the function f_a. The corresponding causal graph is illustrated in Figure 1.

We use an autoencoder to encode the input image into a latent space $\mathbf{L_x}$ and then decode it. We use a VQ-VAE for this autoencoder; more details are given in Section 3.3. We then build a causal model of the transition from $\mathbf{L_x}$ to $\mathbf{L_y}$. This model attempts to learn two components: a vector $\mathbf{a}$ representing the action taken to transition from $\mathbf{L_x}$ to $\mathbf{L_y}$ and an adjacency matrix $\mathbf{M}^{\mathcal{G}}$ representing the causal dependencies in the latent space with respect to this action (the dependencies are not the same, for example, if the action affects the position *vs* the colour of the image). $\mathbf{M}^{\mathcal{G}}$ is specific to an instance but $\mathbf{a}$ models the invariant causal generative mechanisms of the task. In other words, $\mathbf{a}$ represents the *why*, and
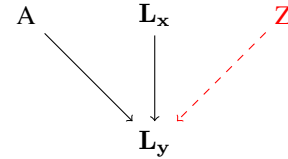

Figure 1: Causal Graph of the transition in latent space. $\mathbf{L_x}$ is the latent representation of the input image X, and $\mathbf{L_y}$ is the latent representation of the output image Y. The transition from X to Y depends on the representations of X and Y and the actions causing the transition. These actions are divided between labelled actions A, which can be controlled, and unknown actions Z, represented by a stochastic process, typically $Z \sim \mathcal{N}(0, 1)$.

$\mathbf{M}^{\mathcal{G}}$ represents the *what* and *how*. A comprehensive description is provided in Section 3.4. Figure 2a shows an overview of our model.

3.3 Multi-Codebook Quantized VAE

We introduce a new autoencoder architecture based on the VQ-VAE called *Multi-Codebook Quantized VAE* or MCQ-VAE. As in [Träuble *et al.*, 2022], our model allows the latent space to have multiple codebooks to increase the expressivity of the quantized vectors. As shown in Figure 2a, each vector is divided into several sub-vectors belonging to a different codebook. In the VQ-VAE, each latent vector embeds local information, e.g. the vector on the top-left corner contains the information needed to reconstruct the top-left corner of the image. Using multiple codebooks allows us to disentangle the local representation into several modules that can be reused across the latent vectors and combined. Each subvector is linked to one causal variable in the causal transition model. The downside of this division into codebooks is memory consumption, which increases linearly with the number of codebooks.

3.4 Latent Causal Transition Model

The autoencoder described in the previous section generates the latent space suitable for our causal transition algorithm. The algorithm can be used in two ways. To apply a transformation on the latent space corresponding to an action with a high-level semantic meaning. Alternatively, given the result of the transformation, to retrieve the corresponding action. To address these two goals and the global reconstruction objective of the autoencoder, we divide our method into three operating modes:

- *Standard:* illustrated in Figure 2b, there is no causal transition, this mode reconstructs the input image X,

- *Action:* illustrated in Figure 2c, a causal transition is applied given an action, the autoencoder must return the image after transition Y,

- *Causal:* illustrated in Figure 2d, given the two images X and Y, before and after transition, the algorithm returns the corresponding action.

Causal Inference The transition from $\mathbf{L_x}$ to $\mathbf{L_y}$ is made using a Graph Neural Network (GNN) architecture where $[\mathbf{L_x}, \mathbf{a}, \mathbf{z}]$ are the nodes and $\mathbf{M}^{\mathcal{G}}$ is the adjacency matrix.
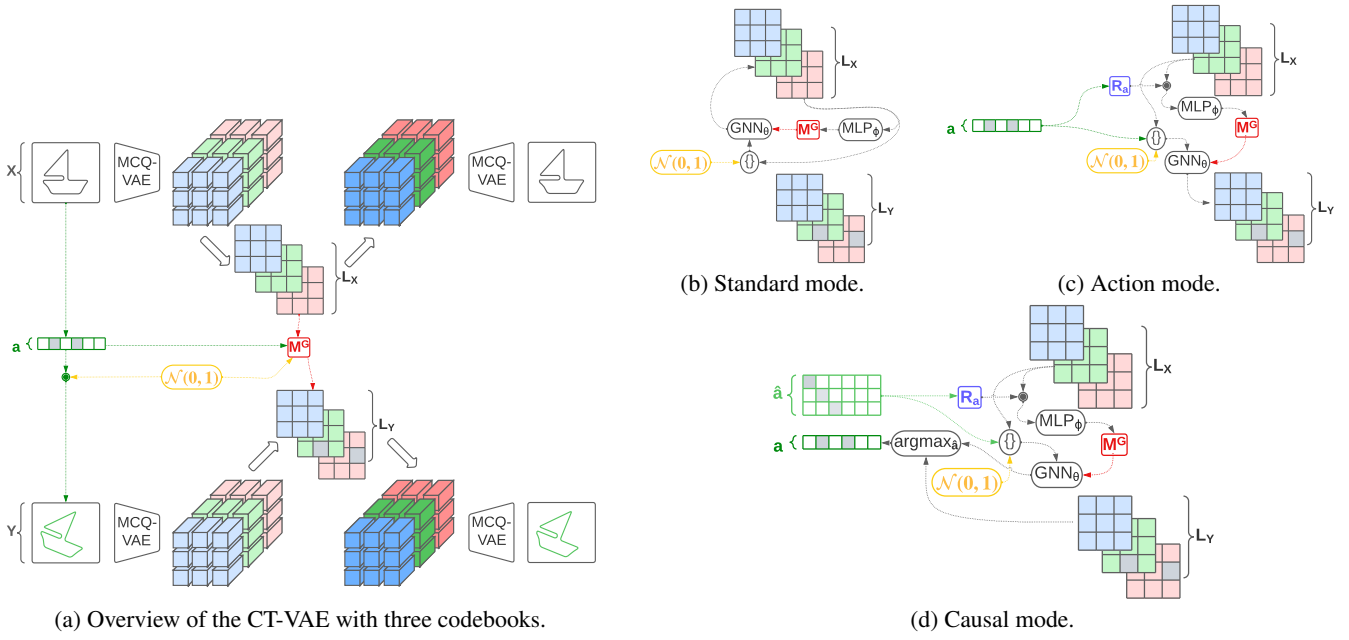
(a) Overview of the CT-VAE with three codebooks.

(b) Standard mode.

(c) Action mode.

(d) Causal mode.

Figure 2: CT-VAE architecture and the three modes of inference. The model is trained to encode and decode an image under an intervention $\mathbf{a}$. The MCQ-VAE generates a quantized latent space and the CT layer performs causal reasoning on that space to modify it according to the intervention. A masked MLP generates the causal graph from the quantized codes under intervention and a GNN infers the corresponding output latent quantized codes from it. In *standard* mode, the CT layer attempts to reproduce the initial space $\mathbf{L_x}$. In *action* mode, it attempts to transpose $\mathbf{L_x}$ to the latent space of the output image $\mathbf{L_y}$. The *causal* mode consists in retrieving the intervention responsible for a transition between X and Y. The action maximising the likelihood of $\mathbf{L_y}$ is selected.

$$\mathbf{L_y} = GNN_\theta([\mathbf{L_x}, \mathbf{a}, \mathbf{z}]; \mathbf{M}^{\mathcal{G}}) \qquad (2)$$

Therefore, the transition problem can be translated to a node classification task. For each variable $L_{xi} \in \mathbf{L_x}$, we aim to find the corresponding L_{yi} based on its parents $pa(L_{yi})$. As shown in Figure 1, the parents of L_{yi} are the action $\mathbf{a}$, a subset of the variables in $\mathbf{L_x}$, and some exogenous variables that are unknown and modelled by a probability distribution $\mathbf{z} \sim \mathcal{N}(0,1)$. The action $\mathbf{a}$ has a global impact that may depend on the node position. To take this into account, we add a positional embedding to each node. The choice of GNNs for the architecture is motivated by their permutation-equivariance property. The second motivation is their ability to model causal graphs where the variables are multi-dimensional [Zecevic *et al.*, 2021].

Causal Structure Discovery The causal graph $\mathcal{G}$ is represented by a learned adjacency matrix $\mathbf{M}^{\mathcal{G}}$. As in previous works [Annadani *et al.*, 2021; Lei *et al.*, 2022], the coefficients α_{ij} of $\mathbf{M}^{\mathcal{G}}$ are obtained using Bernoulli trials with parameters determined by a dense network.

$$\{M_{ij}^{\mathcal{G}}\} \sim \text{Bernouilli}(\sigma(\alpha_{ij}))$$
$$\alpha_{ij} = MLP_\phi([L_{xi}, L_{xj}]; \mathbf{a}) \qquad (3)$$

$\sigma(\cdot)$ is an activation function. We use separate networks for each intervention following the Independent Causal Mechanism (ICM) principle [Schölkopf *et al.*, 2021], which states that the mechanisms involved in data generation do not influence each other. As in [Lei *et al.*, 2022], we use an interven-
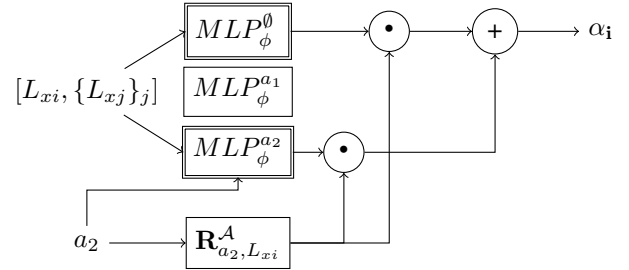

Figure 3: Structure of the causal discovery model in *action* mode. The probability vector $\alpha_\mathbf{i}$ of dependencies of L_{xi} is computed by a dense network with inputs the variable L_{xi} and every other variable L_{xj}. The action a determines which intervention network to use and the mask $\mathbf{R}_{a,L_{xi}}^{\mathcal{A}}$ selects either the intervention network or the network corresponding to no intervention $\emptyset$.

tion mask $\mathbf{R}^{\mathcal{A}}$ to determine which network to use, with $\mathcal{A}$ the set of possible actions $\mathbf{a} \in \mathcal{A}$. Each network computes the probability of existence of a link in the causal graph between two variables L_{xi} and L_{xj} under an intervention from $\mathcal{A}$. $\mathbf{R_a}$ is a binary vector determining whether each causal variable is affected by action $\mathbf{a}$ or not (action $\emptyset$) and selecting the appropriate sub-network as shown on Equation 4 and Figure 3.

$$MLP_\phi([L_{xi}, L_{xj}]; \mathbf{a}) = (MLP_\phi^\mathbf{a}([L_{xi}, L_{xj}]))^{\mathbf{R}_{a,L_{xi}}^{\mathcal{A}}}$$
$$\cdot (MLP_\phi^\emptyset([L_{xi}, L_{xj}]))^{1 - \mathbf{R}_{a,L_{xi}}^{\mathcal{A}}} \qquad (4)$$

We require $|\mathcal{A}| + 1$ networks as we can have $|\mathcal{A}|$ possible

interventions, or no intervention, on a given variable. The intervention mask $\mathbf{R}^{\mathcal{A}}$ is jointly learned with ϕ.

Causal Attribution Finally, in *causal* mode the action $\mathbf{a}$ is obtained by selecting from the set $\mathcal{A}$ of possible actions the one corresponding to the optimal transition to $\mathbf{L_y}$.

$$\mathbf{a} = \underset{\hat{\mathbf{a}} \in \mathcal{A}}{\mathrm{argmax}} \left(\mathbb{E}_{\mathbf{L_y}}[GNN_\theta([\mathbf{L_x}, \hat{\mathbf{a}}, \mathbf{z}]; \mathbf{M}^{\mathcal{G}})] \right)$$

$$\text{with } \mathbf{M}^{\mathcal{G}} \sim \mathrm{Bernouilli}(\sigma(MLP_\phi([\mathbf{L_x}]; \hat{\mathbf{a}}))) \quad (5)$$

3.5 Training

The model is trained in two stages. First, we pre-train the MCQ-VAE on a reconstruction task using the same procedure as in the original VQ-VAE. Second, we plug the Causal Transition layer into the architecture and train it on the transition task. The weights of the MCQ-VAE are frozen during this stage. Several losses and regularisation methods are added to help the model perform causal transition. During this stage, the learning process is divided further into three alternating steps. These steps correspond to the *standard*, *action*, and *causal* modes.

Standard In *standard* mode, the transition model must behave like the identity function, as shown in Figure 2b. Given $\mathbf{L_x}$ and the *null* action $\emptyset$, the causal graph $\mathbf{M}^{\mathcal{G}}$ should be the identity matrix $\mathbf{I}$ and $\mathbf{L_y}$ equals $\mathbf{L_x}$.

$$\mathcal{L}_x(\phi, \theta) = \mathbb{E}_{\mathbf{L_x}}[GNN_\theta([\mathbf{L_x}, \mathbf{z}]; \mathbf{M}^{\mathcal{G}})]$$

$$\text{with } \mathbf{M}^{\mathcal{G}} \sim \mathrm{Bernoulli}(\sigma(MLP_\phi([\mathbf{L_x}]; \emptyset))) \quad (6)$$

The primary loss function used is represented on Equation 6. It maximizes the likelihood of the generated representation, driving it towards $\mathbf{L_x}$. In addition to this loss function, two regularisation losses are used.

$$\mathcal{L}_{id_y}(\theta) = \mathbb{E}_{\mathbf{L_x}}[GNN_\theta([\mathbf{L_x}, \mathbf{z}]; \mathbf{I})] \quad (7)$$

$$\mathcal{L}_{id_{M^{\mathcal{G}}}}(\phi) = \|\mathrm{Bernoulli}(\sigma(MLP_\phi([\mathbf{L_x}]; \emptyset)))) - \mathbf{I}\|^2 \quad (8)$$

The loss function in Equation 7 maximizes the likelihood of the output of the GNN parameterised by θ given a causal graph being equal to the identity, and the one in Equation 8 regularises $\mathbf{M}^{\mathcal{G}}$ and the parameters ϕ towards the identity matrix. As in [Lei *et al.*, 2022], the Straight-Through Gumbel-Softmax reparametrisation trick [Jang *et al.*, 2017] is used to allow the gradient to flow through the Bernoulli sampling. The last two losses are only used in *base* mode.

Action In *action* mode, the transition model must transform $\mathbf{L_x}$ into $\mathbf{L_y}$, as shown in Figure 2c. This is a two-steps process. First, given $\mathbf{L_x}$ and $\mathbf{a}$, the model learns $\mathbf{M}^{\mathcal{G}}$. Second, given $\mathbf{L_x}$, $\mathbf{a}$, and $\mathbf{M}^{\mathcal{G}}$, the model infers $\mathbf{L_y}$.

$$\mathcal{L}_y(\phi, \theta) = \mathbb{E}_{\mathbf{L_y}}[GNN_\theta([\mathbf{L_x}, \mathbf{a}, \mathbf{z}]; \mathbf{M}^{\mathcal{G}})]$$

$$\text{with } \mathbf{M}^{\mathcal{G}} \sim \mathrm{Bernoulli}(\sigma(MLP_\phi([\mathbf{L_x}]; \mathbf{a}))) \quad (9)$$

The loss function in Equation 9 ensures that the transition model parameterized by θ and ϕ accurately generates $\mathbf{L_y}$ in *action* mode. The Straight-Through Gumbel-Softmax [Jang *et al.*, 2017] reparametrisation trick is used again. This loss function is identical to the first one introduced in *standard* mode, but given an intervention.

Causal In *causal* mode, the model does not output a reconstructed image but an action vector, as shown in Figure 2d. The decoder is not called, instead we introduce a loss function maximising the likelihood of the generated action vector.

$$\mathcal{L}_\mathbf{a}(\phi, \theta) = \mathbb{E}_\mathbf{a}[\mathbf{q}]$$

$$\text{with } q_{\hat{\mathbf{a}}} = \frac{e^{\mathbb{E}_{\mathbf{L_y}}[GNN_\theta([\mathbf{L_x}, \hat{\mathbf{a}}, \mathbf{z}]; \mathbf{M}^{\mathcal{G}}(\hat{\mathbf{a}}))]}}{\sum\limits_{\mathbf{a} \in \mathcal{A}} e^{\mathbb{E}_{\mathbf{L_y}}[GNN_\theta([\mathbf{L_x}, \mathbf{a}, \mathbf{z}]; \mathbf{M}^{\mathcal{G}}(\mathbf{a}))]}}$$

$$\text{and } \mathbf{M}^{\mathcal{G}}(\mathbf{a}) \sim \mathrm{Bernouilli}(\sigma(MLP_\phi([\mathbf{L_x}]; \mathbf{a}))) \quad (10)$$

The output in *causal* mode is a vector $\mathbf{q} \in \mathbb{R}^{|\mathcal{A}|}$ corresponding to the probability for each action to be the cause of the transition from $\mathbf{L_x}$ to $\mathbf{L_y}$. It is obtained by computing the transition in *action* mode for each action in $\mathcal{A}$ and its likelihood given the true $\mathbf{L_y}$. The likelihoods are converted to probabilities using softmax activation. The resulting vector $\mathbf{q}$ is trained to resemble the true action vector $\mathbf{a}$ using the loss $\mathcal{L}_\mathbf{a}(\phi, \theta)$.

Graph Regularisation The likelihood of the output $\mathbf{L_y}$ cannot be directly optimised because the causal graph $\mathcal{G}$ is unknown and acts as a latent variable. In consequence, we maximise the Evidence Lower Bound (ELBO) shown in Equation 11, as in VCN and VCD [Annadani *et al.*, 2021; Lei *et al.*, 2022].

$$\mathbb{ELBO}(\phi, \theta) = \mathbb{E}_{q_\phi(\mathcal{G})}[p_\theta(\mathbf{L_y}|\mathcal{G})] - \mathrm{KL}(q_\phi(\mathcal{G})\|p(\mathcal{G})) \quad (11)$$

The first term is the reconstruction term corresponding to the losses $\mathcal{L}_x$ and $\mathcal{L}_y$ introduced previously in this section. We now derive the regularisation term $\mathrm{KL}(q_\phi(\mathcal{G})\|p(\mathcal{G}))$ where KL is the Kullback–Leibler divergence, $p(\mathcal{G})$ is the prior distribution over causal graphs, and $q_\phi(\mathcal{G})$ is the learned posterior distribution, as shown on Equations 3 and 12.

$$q_\phi(M_{ij}^{\mathcal{G}}) = q_\phi(\alpha_{ij}|\mathbf{a}, L_{xi}, L_{xj}) = \sigma(\alpha_{ij}) \quad (12)$$

Unlike in VCN, we do not need to constrain the space of graphs to Directed Acyclic Graphs (DAGs), as our graph is a DAG by construction. All edges start in the set $\mathbf{L_x}$ and end in the set $\mathbf{L_y}$. Thus, the prior probability of the existence of an edge follows the uniform law:

$$p(\mathbf{M}_{ij}^{\mathcal{G}}) = \mathrm{Uniform}(0, 1) \quad (13)$$

To help regularise the causal graph $\mathcal{G}$ further, two other losses are introduced for the set of parameters ϕ.

$$\mathcal{L}_{|M^{\mathcal{G}}|}(\phi) = \|\mathrm{Bernoulli}(\sigma(MLP_\phi([\mathbf{L_x}]; \mathbf{a})))\|^2 \quad (14)$$

Equation 14 reduces the norm of the generated causal graph and, by extension, minimises the number of dependencies of the causal variables.

$$\mathcal{L}_{dep(M^{\mathcal{G}})}(\phi) = \sum_i \|\prod_j (1 - \sigma(MLP_\phi([L_{xi}, L_{xj}]; \mathbf{a})))\|^2$$

$$(15)$$

Finally, Equation 15 minimises, for each node, the joint probability of having no dependencies, and ensures that at least one dependency will exist for every node of the graph.

4 Experiments

4.1 Datasets

We perform our experiments on several standard disentanglement benchmarks. The Cars3D dataset [Reed *et al.*, 2015] contains 3D CAD models of cars with 3 factors of variation: the type of the car, camera elevation, and azimuth. The Shapes3D dataset [Kim and Mnih, 2018] contains generated scenes representing an object standing on the floor in the middle of a room with four walls. The scene contains 6 factors of variation: the floor, wall and object colours, the scale and shape of the object, and the orientation of the camera in the room. The Sprites dataset [Reed *et al.*, 2015] contains images of animated characters. There are 9 variant factors corresponding to character attributes such as hair or garments. The DSprites dataset [Higgins *et al.*, 2017] contains 2D sprites generated based on 6 factors: the colour, shape, and scale of the sprite, the location of the sprite with x and y coordinates, and the rotation of the sprite.

All the datasets described above are synthetic, and all of their generative factors of variation are labelled. We also apply our model to real-world data. The CelebA dataset [Liu *et al.*, 2015] is a set of celebrity faces labelled with 40 attributes including gender, hair colour, and age. Unlike the above datasets, these attributes do not fully characterise each image. Many attributes linked to the morphology of the face are not captured or are captured with insufficient precision to uniquely correspond to an individual. These missing attributes correspond to exogenous factors of variation.

We build the transitions (X, Y) using the given factors of variation. For instance, two images X and Y can be part of a transition if all their factors are identical but one. We generate the transitions (X, Y) and (Y, X) with two opposite actions a and $-a$ updating the value of the corresponding factor of variation.

Imbalanced Data Figure 4 shows the distribution of actions in the datasets. The factors are highly unbalanced for every dataset. For instance, the Cars3D dataset has three factors of variation. The first one (in green) has few variations in the distribution, the second (in blue) has six times more variations, and the third one (in red) has thirty times more variations. The data are not i.i.d. To tackle this issue, we adopt a model-centric approach powered by causality theory: the causal graph built by our model aims to eliminate the effect of the spurious correlations induced by the data imbalance. Our experiments show that our model can learn to distinguish the factors of variation efficiently, and significantly reduces the effect of confounders.

4.2 Image Generation Under Intervention

We perform a series of interventions on input images and study the quality of the generated images. After each intervention, we take the result of generation and apply a new intervention to it. Figure 5 illustrates the result for the Shapes3D dataset. We can observe that the reconstructed images do not undergo a quality reduction. This is expected as our method does not affect the codebooks created by vector quantization. We can also see that, after intervention, the reconstructed images have only the intervened-on factor modi-
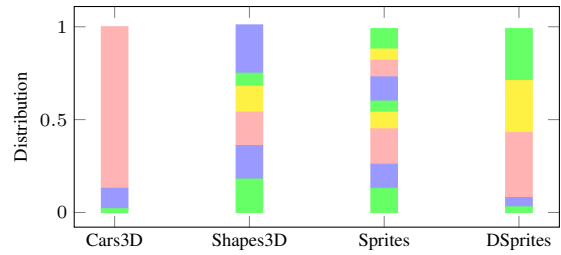

Figure 4: Distribution of the factors of variation for each dataset. The longer the bar the higher the number of variations for the corresponding factor.
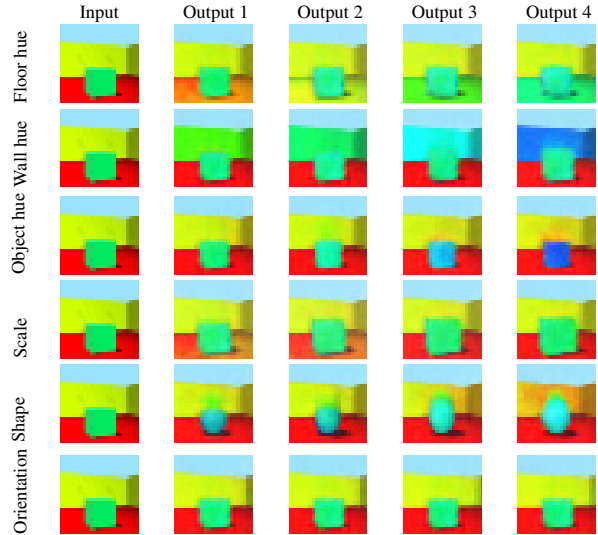

Figure 5: Atomic interventions on one factor of variation on images from the Shapes3D dataset. Each row corresponds to an intervention on a different factor with the first row being the input image. Output i corresponds to the output after applying the same action i times.

fied. For example, background colours are not modified when operating on the object colour. Similarly, the more complex intervention on the camera orientation involves many changes in the pixels of the image but is correctly handled by the CT-VAE. Therefore, our method can properly disentangle the factors of variation and discriminate among the variables affected and unaffected by the intervention. We can be observe a few exceptions. Changing the shape of the object generates slight modifications of the background near the object. As we use the output for the next generation, these modifications may propagate. Further studies and results for the other datasets are given in the appendix.

4.3 Causal Structure Discovery

We now look at the structure of our causal transition model. Figure 6 shows the generated latent adjacency matrices and the causal masks. The dependencies are very different depending on the dataset on which the model was trained. In the Cars3D dataset, the variables mainly look at the bottom half of the latent image. The nature of the car images can explain this behaviour; all the Cars3D cars are located in the
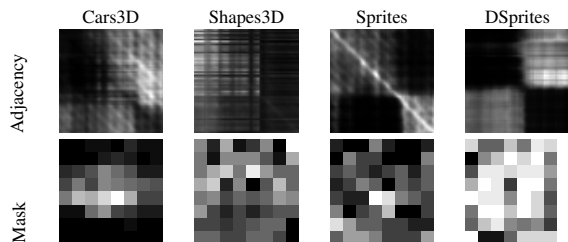
Figure 6: Causal structure of the CT layer for each dataset. Masks are $H \times W$ matrices with H and W respectively the height and width of the latent image. Adjacency matrices are $HW \times HW$ matrices. Adjacencies and masks are averaged over a batch. The brightness indicates the probability of existence of a link between two variables for the adjacency matrix. For the mask, it indicates the probability that the variable undergoes an intervention.

	Cars3D	Shapes3D	Sprites	DSprites
#Actions	6	12	18	10
Action Acc.	0.71	0.83	0.42	0.44
Factor Acc.	0.94	0.90	0.82	0.58

Table 1: Accuracy of *causal* mode. The task aims to retrieve the action causing the transition from image X to Y. *#Actions* shows the number of possible action, i.e. the cardinality of the output space. Each factor of variation has two associated actions, one increasing and the other decreasing its value. *Action Acc.* shows the action accuracy and *Factor Acc.* shows the factor accuracy.

middle, on a uniform background. The car can be slightly below the middle, depending on the camera orientation. Thus, the elements of the image affected by an intervention are also in the middle of the image. This behaviour is consistent with the masked image, which shows that the zones where the intervention takes place match the possible locations of the car. This behaviour is not observed for the Shapes3D dataset, where the background is also affected by interventions.

Action Recovery As detailed in Section 4.2, the CT-VAE supports interventions affecting a single factor of variation. Given the causal graph, we would like to see whether this factor can be recovered. A factor of variation has a value evolving along an axis, either increasing or decreasing. We represent actions as one-hot vectors, so increments and decrements are considered different actions. We consider the problem of recovering the factor of variation, and the problem of recovering the action, i.e. the factor of variation and the direction. Table 1 summarises the results. The CT-VAE can retrieve with high accuracy the correct action for the Cars3D and Shapes3D datasets but struggle with Sprites and DSprites, which contain smaller sprites than the former datasets. For the Sprites dataset, the model has trouble identifying the direction of the action but can retrieve the correct factor in most cases. We can observe that the number of actions has little impact on the accuracy.

5 Discussion and Conclusion

Recovering the mechanisms generating images is a challenging task. Current disentanglement methods rely on Variational Auto-Encoders and attempt to represent the various factors of variation responsible for data generation on separate dimensions. We propose a new method based on causality theory to perform disentanglement on quantized VAEs. Our method can perform interventions on the latent space affecting a single factor of variation. We test it on synthetic and real-world data.

A limitation of our current architecture is the division between the pre-training and fine-tuning stages. Codebooks are fixed in the fine-tuning stage, limiting the CT layer in both the level of expressivity and the disentanglement of latent codes. Training the two parts of the model jointly on a reconstruction and transition task could alleviate this issue but would require regularising the distribution of latent codes. Our model is also limited by the set of actions, which must be known in advance. In future work, we will attempt to solve these issues, including learning the set of possible actions.

One of the questions that our method raises regards the level of disentanglement of the latent space. The latent space studied in this paper is of a very different nature from the ones studied in the standard disentanglement literature. The VAE latent space is traditionally a $\mathbb{R}^D$ vector where each dimension accounts for one factor of variation if accurately disentangled. The disentanglement ability of our model comes from its accurate identification of the relevant latent variables subject to intervention in the causal graph when one factor of variation is modified. This difference, unfortunately, prevents us from comparing the level of disentanglement of our model using standard metrics like DCI [Eastwood and Williams, 2018] or SAP [Kumar *et al.*, 2018]. We leave the question of developing precise disentanglement measures for quantized latent spaces for future work.

References

[Ahuja *et al.*, 2022] Kartik Ahuja, Yixin Wang, Divyat Mahajan, and Yoshua Bengio. Interventional causal representation learning. *CoRR*, abs/2209.11924, 2022.

[Annadani *et al.*, 2021] Yashas Annadani, Jonas Rothfuss, Alexandre Lacoste, Nino Scherrer, Anirudh Goyal, Yoshua Bengio, and Stefan Bauer. Variational causal networks: Approximate bayesian inference over causal structures. *CoRR*, abs/2106.07635, 2021.

[Bareinboim *et al.*, 2022] Elias Bareinboim, Juan D Correa, Duligur Ibeling, and Thomas Icard. On pearl's hierarchy and the foundations of causal inference. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, pages 507–556. 2022.

[Chen *et al.*, 2018] Tian Qi Chen, Xuechen Li, Roger B. Grosse, and David Duvenaud. Isolating sources of disentanglement in variational autoencoders. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 2615–2625, 2018.

[Eastwood and Williams, 2018] Cian Eastwood and Christopher K. I. Williams. A framework for the quantitative

evaluation of disentangled representations. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[Gabbay *et al.*, 2021] Aviv Gabbay, Niv Cohen, and Yedid Hoshen. An image is worth more than a thousand words: Towards disentanglement in the wild. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 9216–9228, 2021.

[Geffner *et al.*, 2022] Tomas Geffner, Javier Antoran, Adam Foster, Wenbo Gong, Chao Ma, Emre Kiciman, Amit Sharma, Angus Lamb, Martin Kukla, Nick Pawlowski, Miltiadis Allamanis, and Cheng Zhang. Deep end-to-end causal inference. *CoRR*, abs/2202.02195, 2022.

[Gresele *et al.*, 2021] Luigi Gresele, Julius von Kügelgen, Vincent Stimper, Bernhard Schölkopf, and Michel Besserve. Independent mechanism analysis, a new concept? In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 28233–28248, 2021.

[Hamilton *et al.*, 2017] William L. Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1024–1034, 2017.

[Higgins *et al.*, 2017] Irina Higgins, Loïc Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, Matthew M. Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[Higgins *et al.*, 2018] Irina Higgins, David Amos, David Pfau, Sébastien Racanière, Loïc Matthey, Danilo J. Rezende, and Alexander Lerchner. Towards a definition of disentangled representations. *CoRR*, abs/1812.02230, 2018.

[Horan *et al.*, 2021] Daniella Horan, Eitan Richardson, and Yair Weiss. When is unsupervised disentanglement possible? In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 5150–5161, 2021.

[Jang *et al.*, 2017] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[Kim and Mnih, 2018] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 2654–2663. PMLR, 2018.

[Kingma and Welling, 2014] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[Kipf and Welling, 2017] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[Kumar *et al.*, 2018] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational inference of disentangled latent concepts from unlabeled observations. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[Lachapelle *et al.*, 2022] Sébastien Lachapelle, Pau Rodríguez, Yash Sharma, Katie Everett, Rémi Le Priol, Alexandre Lacoste, and Simon Lacoste-Julien. Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ICA. In Bernhard Schölkopf, Caroline Uhler, and Kun Zhang, editors, *1st Conference on Causal Learning and Reasoning, CLeaR 2022, Sequoia Conference Center, Eureka, CA, USA, 11-13 April, 2022*, volume 177 of *Proceedings of Machine Learning Research*, pages 428–484. PMLR, 2022.

[Lei *et al.*, 2022] Anson Lei, Bernhard Schölkopf, and Ingmar Posner. Causal discovery for modular world models. In *NeurIPS 2022 Workshop on Neuro Causal and Symbolic AI (nCSI)*, 2022.

[Liu *et al.*, 2015] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 3730–3738. IEEE Computer Society, 2015.

[Locatello *et al.*, 2019] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In Kamalika Chaudhuri and Ruslan

Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 4114–4124. PMLR, 2019.

[Locatello *et al.*, 2020] Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 6348–6359. PMLR, 2020.

[Mathieu *et al.*, 2019] Emile Mathieu, Tom Rainforth, N. Siddharth, and Yee Whye Teh. Disentangling disentanglement in variational autoencoders. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 4402–4412. PMLR, 2019.

[Pearl, 2009] Judea Pearl. *Causality*. Cambridge university press, 2009.

[Ramesh *et al.*, 2021] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8821–8831. PMLR, 2021.

[Razavi *et al.*, 2019] Ali Razavi, Aäron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 14837–14847, 2019.

[Reed *et al.*, 2015] Scott E. Reed, Yi Zhang, Yuting Zhang, and Honglak Lee. Deep visual analogy-making. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1252–1260, 2015.

[Sánchez-Martín *et al.*, 2022] Pablo Sánchez-Martín, Miriam Rateike, and Isabel Valera. VACA: designing variational graph autoencoders for causal queries. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 8159–8168. AAAI Press, 2022.

[Schölkopf *et al.*, 2021] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proc. IEEE*, 109(5):612–634, 2021.

[Suter *et al.*, 2019] Raphael Suter, Đorđe Miladinovic, Bernhard Schölkopf, and Stefan Bauer. Robustly disentangled causal mechanisms: Validating deep representations for interventional robustness. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 6056–6065. PMLR, 2019.

[Träuble *et al.*, 2022] Frederik Träuble, Anirudh Goyal, Nasim Rahaman, Michael Mozer, Kenji Kawaguchi, Yoshua Bengio, and Bernhard Schölkopf. Discrete key-value bottleneck. *CoRR*, abs/2207.11240, 2022.

[van den Oord *et al.*, 2017] Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6306–6315, 2017.

[Velickovic *et al.*, 2018] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[Williams *et al.*, 2021] Jennifer Williams, Yi Zhao, Erica Cooper, and Junichi Yamagishi. Learning disentangled phone and speaker representations in a semi-supervised VQ-VAE paradigm. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2021, Toronto, ON, Canada, June 6-11, 2021*, pages 7053–7057. IEEE, 2021.

[Yang *et al.*, 2021] Mengyue Yang, Furui Liu, Zhitang Chen, Xinwei Shen, Jianye Hao, and Jun Wang. Causalvae: Disentangled representation learning via neural structural causal models. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 9593–9602. Computer Vision Foundation / IEEE, 2021.

[Zecevic *et al.*, 2021] Matej Zecevic, Devendra Singh Dhami, Petar Velickovic, and Kristian Kersting. Relating graph neural networks to structural causal models. *CoRR*, abs/2109.04173, 2021.