

# Interactive Disentanglement: Learning Concepts by Interacting with their Prototype Representations

Wolfgang Stammer<sup>1</sup>Marius Memmel<sup>1</sup>Patrick Schramowski<sup>1</sup>Kristian Kersting<sup>1,2</sup><sup>1</sup>Technical University of Darmstadt, Computer Science Department<sup>2</sup>Technical University of Darmstadt, Centre for Cognitive Science

{wolfgang.stammer@cs, marius.mommel@stud, schramowski@cs, kersting@cs}.tu-darmstadt.de

December 7, 2021

## Abstract

Learning visual concepts from raw images without strong supervision is a challenging task. In this work, we show the advantages of prototype representations for understanding and revising the latent space of neural concept learners. For this purpose, we introduce interactive Concept Swapping Networks (iCSNs), a novel framework for learning concept-grounded representations via weak supervision and implicit prototype representations. iCSNs learn to bind conceptual information to specific prototype slots by swapping the latent representations of paired images. This semantically grounded and discrete latent space facilitates human understanding and human-machine interaction. We support this claim by conducting experiments on our novel data set “Elementary Concept Reasoning” (ECR), focusing on visual concepts shared by geometric objects.

## 1 Introduction

Learning an adequate representation of concepts from raw data without strong supervision is a challenging task. However, it remains important for research in areas of knowledge discovery where sufficient prior knowledge is missing, and the goal is to attain novel understandings. With better representations and architectural components of machine learning models, this appears to become more and more achievable [71]. However, if remained unchecked this bears the danger of learning incorrect concepts or even confounding features [18, 68]. A further difficult aspect of concept learning, regardless of the level of supervision, is its dynamic and subjective nature. One downstream task might require more fine-grained concepts than others, but also when encountering evidence on novel concepts (*e.g.* in an online learning setting), the knowledge and hierarchy of concepts should be constantly re-approved, discussed, and possibly updated. It thus remains desirable that the representations learned by such concept learners to be human-understandable and revisable.

An evident approach to teaching concept information to a machine learning model is to train it in a supervised fashion through symbolic representations, *e.g.*, one-hot encoding vectors and corresponding raw input [83, 42]. However, this requires extensive prior knowledge of relevant concepts and seems impractical given the subjective and dynamic nature of

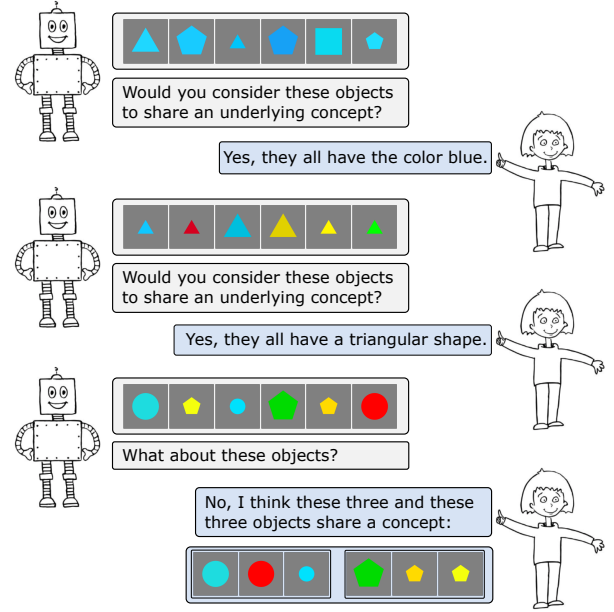


Figure 1: A trained model (left) queries the human user (right) if the concepts that it has extracted from the data coincides with the knowledge of the user. Subsequently, the model can receive revisions from the user.

concept learning.

Another branch of research focuses on learning disentangled latent distribution models [25, 29, 78]. Although initially focused on unsupervised approaches, many recent studies have shifted away from unsupervised learning and show promising results with weak supervision [47, 72, 49, 79, 56, 39]. An often implicit assumption of disentanglement research is that the learned latent representations should correspond to human-interpretable factors. Many state-of-the-art variational [49, 56] and generative adversarial [13, 58, 57, 44, 54] approaches, however, learn continuous latent representations, making these difficult for a human to understand without additional techniques for interpreting the latent space [66].

For handling the intricate nature of concept learning and inspired by findings on concept prototypes in the fields of psychology and cognitive science, we investigate the advantages of prototype representations in learning human-understandable and revisable concept representations for neural concept learners. To this end, we introduce the novel frame-

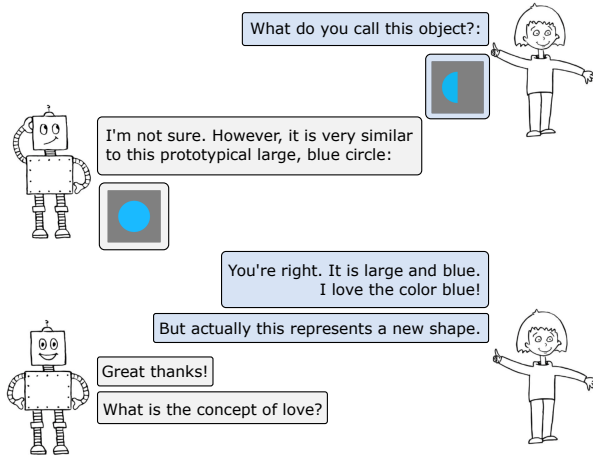


Figure 2: Human-machine interaction for learning about novel concepts. The user queries an object and guides the machine’s prototype suggestion if necessary.

work of Interactive Concept Swapping Network (iCSN) that learns to implicitly bind semantic concepts to latent prototype representations through weak supervision. This binding is enforced via a discretized distance estimation and swapping of shared concept representations between paired data samples. Among other things, iCSNs allow for querying and revising its learned concepts *cf.* 1, and for easily integrating knowledge of unseen concepts *cf.* 2.

Explicitly focusing on learning object-centric visual concepts, we develop Elementary Concept Reasoning (ECR), a novel data set containing images of 2D geometrical objects and perform multiple experiments, emphasizing the advantages of our approach. To summarize, our work highlights the advantages of prototype representations for (i) learning a consistent and human-understandable latent space through weak supervision, (ii) revising concept representations via human interactions, and (iii) updating these in an online learning fashion.

## 2 Related Work

**Concept learning.** Many previous concept learning approaches focus on predicting selected high-level concepts for improving additional downstream tasks [83, 42, 3]. Several studies highlight the benefits of concept-based machine learning for explainability [1, 82, 23, 52] and human interactions [75]. To communicate the concepts to a human user, some approaches include first-order logic formulas [16], causal relationships [81], user defined concepts [40], prediction of intermediate dataset-labels [3, 52], and one-hot encoded bottlenecks [42]. All of these approaches, however, focus on supervised concept learning.

**Concept representations in psychology.** The term concept is rooted in psychology where it can be defined as “*the label of a set of things that have something in common*” [2], though different notions do exist [21]. Most common approaches to represent concepts are exemplars and prototypes. Where the former approach assumes that one or multiple typical examples of a concept are maintained in memory, the latter only assumes an average representation over several observed examples [69, 59, 76, 73, 37]. The lines between exemplar and

prototype representation become more blurred in the field of cognitive psychology, and their contribution to concept representations is still an open problem, with recent work hinting at the use of both representations [6, 22]. Nonetheless, there remains evidence of the use and importance of prototypes in the human memory system [17, 28, 36]. Inspiration sparked by such findings gave rise to Prototype Learning Systems [85].

**Neural prototype learning.** Recent approaches to artificial prototype learning systems focus on neural networks with prototype vectors as internal latent representations. These vectors can be converted into explainable visualizations via decoding approaches [43] or used for finding the most similar training example [11]. In both works, a class prediction is made based on the similarity of an encoded input to the model’s prototypes via a simple distance metric. Lastly, especially in the context of few-shot learning, prototypes show advantageous properties [74, 62, 60, 35].

**Disentanglement.** The field of disentanglement research is also closely related to our work. Here the goal is to extract independent underlying factors that are responsible for generating the data [4]. Recently, through the work of Locatello *et al.* [47] much of disentanglement research has shifted from unsupervised learning to the weakly supervised setting. Shu *et al.* [72] show that supervision via match pairing for a known subset of factors gives guarantees for disentanglement via their defined calculus of consistency and restrictiveness. Literature also extends to group-based disentanglement, allowing for grouping of the identified generative factors [32, 31, 86, 5, 34, 84]. However, the interpretation of the latent representations from these approaches remains an open question [66].

**Explanatory interactive learning (XIL).** The notion of human interactions on a model’s latent concept representations, *e.g.*, to correct confounding behaviour, is closely related to the field of XIL [67, 70, 77, 68, 75]. Specifically, XIL incorporates the human user into the training loop by allowing for them to interact via a model’s explanations. Rather than interacting via post-hoc explanations of previous XIL approaches, we focus on interacting directly with the latent representations of a model. These, nonetheless, possess a connection to the model’s explanations in our setup. Even though we take a more direct approach to revising a model’s internal representations, similar feedback methods as in XIL are applicable for our work.

## 3 Interactive Concept Swapping Networks

In this section, we explain the basic architectural components of an Interactive Concept Swapping Network (iCSN) before introducing the training procedure and how to interact with the implicit prototype representations of these networks. For an overview, see 3.

**Prototype-based concept architecture.** Assume an input  $x_i \in X$ , with  $X := [x_1, \dots, x_N] \in \mathbb{R}^{N \times D}$ . For the sake of simplicity, we remove the sample index  $i$  from further notations below and denote with  $x \in \mathbb{R}^D$  an entire image. However, in our framework,  $x$  can also be a latent representation of a subregion of the image. This subregion can be implic-

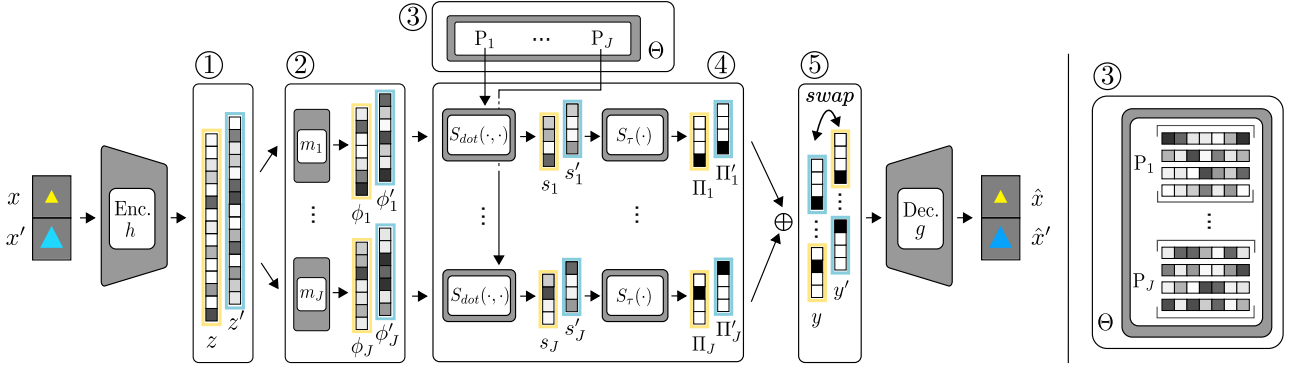


Figure 3: Interactive Concept Swapping Network. An iCSN is based on a deterministic autoencoder structure providing an initially entangled latent encoding (1). Several read-out encoders (2) extract relevant information from this latent space and compare their extracted concept encodings to a set of prototype slots (3) via a weighted, softmax-based dot product (4). This leads to a discretized code that indicates the most similar prototype slot of each concept encoding. iCSNs are trained via a simple reconstruction loss, weak supervision via match pairing and a swapping approach that swaps (5) the latent concept representations for shared concepts, enforcing the binding of semantic information to specific prototype representations.

itly or explicitly extracted from the image by a pre-processing step, *e.g.* via segmentation algorithms [24, 27, 64, 8] or compositional generative scene models [7, 19, 26, 46, 50]. Additionally, we assume each  $x$  to contain several attributes such as color, shape and size. Specifically, we refer to the realizations of these attributes, *e.g.*, a “blue color” or “triangular shape” as a *basic concept*. In contrast, we refer to “color” as a category concept or, as often called in the field of cognitive and psychological sciences, *superordinate concept* [20]. Each image  $x$  therefore has the ground truth basic concepts  $c := [c_1, \dots, c_J]$  with  $J$  denoting the total number of superordinate concepts. We make the necessary assumption that  $x$  can only contain one basic concept realization per superordinate concept. For simplicity, we furthermore assume that each superordinate concept contains the same number of basic concepts  $K$  which might vary in practice as we are going to show in our experiments.

Assuming an encoder-decoder structure, we define an input encoder  $h(\cdot)$  that receives the input image  $x$  and encodes it into a latent representation  $z \in \mathbb{R}^Z$  by  $h(x) = z$ . Rather than reconstructing directly from  $z$ , as done by many autoencoder-based approaches, an iCSN first applies several read-out encoders  $m_j(\cdot)$  to the latent representation  $z$  resulting in  $m_j(z) = \phi_j \in \mathbb{R}^Q$  with  $j \in [1, \dots, J]$ . We refer to the resulting encoding  $\phi_j$  as a *concept encoding*. The goal of each read-out encoder is to extract the relevant information from the entangled latent space  $z$  that corresponds to a superordinate concept, *e.g.* color. Below we are going to discuss how we enforce this extraction of concept specific information.

One central component of the iCSN is a set of codebooks each containing multiple prototype slots. We define this set as  $\Theta := [P_1, \dots, P_J]$ , with a single codebook as  $P_j \in \mathbb{R}^{Q \times K}$ . Each codebook contains an ordered set of trainable, randomly initialized prototype slots  $p_j \in \mathbb{R}^Q$ , *i.e.*,  $P_j := [p_j^1, \dots, p_j^K]$ .

To enforce the assignment of each concept encoding  $\phi_j$  to one prototype slot of  $P_j$ , we define a similarity score  $S_{dot}(\cdot, \cdot)$  as a softmax over the dot product between its two inputs. This way we obtain the similarity between a concept encoding,  $\phi_j$ ,

and specific prototype slot,  $p_j^k$  with:

$$s_j^k = S_{dot}(\phi_j, p_j^k) = \frac{\exp(\phi_j \cdot p_j^k / \sqrt{Q})}{\sum_{k=1}^K \exp(\phi_j \cdot p_j^k / \sqrt{Q})} \quad (1)$$

The resulting similarity vector  $s_j \in \mathbb{R}^K$  contains the similarity score for each prototype slot of category  $j$  with the concept encoding  $\phi_j$ . To enforce further discretization and the binding of concepts to individual prototype slots, we introduce a second function  $S_\tau(\cdot)$  to apply a weighted softmax function to the similarity scores:

$$\Pi_j^k = S_\tau(s_j^k) = \frac{\exp(s_j^k / \tau)}{\sum_{k=1}^K \exp(s_j^k / \tau)}, \quad (2)$$

with  $\Pi_j \in \mathbb{R}^K$  and weight parameter  $\tau \in \mathbb{R}^+$ . In our experiments we step-wise decrease  $\tau$  to gradually enforce the binding of information. In the extreme case of  $\tau \rightarrow 0$ ,  $\Pi_j$  resembles a one-hot vector (multi-label one-hot vector in the case of  $J > 1$ ), indicating which prototype slot of category  $j$  the concept encoding  $\phi_j$  is most similar to.

Finally, we concatenate the weighted similarity scores of each category into a single vector to receive the final prototype distance codes  $y := [\Pi_1, \dots, \Pi_J] \in [0, 1]^{J \cdot K}$  which we pass to the decoder  $g(\cdot)$  to reconstruct the image:  $g(y) = \hat{x} \in \mathbb{R}^D$ .

**Concept swapping and weak supervision.** Prior to training, *i.e.*, after initialization, there is no semantic knowledge bound to the prototype slots yet. Each prototype carries just as little meaning as the other. The semantic knowledge found in converged iCSNs, however, is indirectly learned via a weakly-supervised training procedure and by employing a simple swapping trick.

We adopt the match pairing approach of Shu *et al.* [72], a practical weakly-supervised training procedure to overcome the issues of unsupervised disentanglement [48]. In this approach, a pair of images  $(x, x')$  is observed that shares values for a known subset of underlying factors of variation within the data, *e.g.* color, while the total number of shared factors can vary between 1 and  $J - 1$ . In this way, a model can use the additional information of the pairing to constrain and guide the learning of its latent representations.

Previous works on weakly-supervised training, specifically of VAEs, reverted to applying a product [5] or an average [33] of the encoder distributions of  $x$  and  $x'$  at the shared factor IDs. Locatello *et al.* [49], extended these works to a setting with an even weaker form of supervision but carries fewer disentanglement guarantees. In comparison to these works, an iCSN uses a simple swapping trick between paired representations, similar to Caron *et al.* [9]. Specifically, with  $v$  being the shared factor ID between the image pairs  $(x, x')$  the corresponding similarity scores  $(\Pi_v, \Pi'_v)$  are swapped between the final corresponding prototype codes, resulting in:

$$y := [\Pi_1, \dots, \Pi'_v, \dots, \Pi_J], y' := [\Pi'_1, \dots, \Pi_v, \dots, \Pi'_J].$$

This swapping procedure has the intuitive semantic that it forces an iCSN to extract information from the first image that it can use to represent properties of the category  $v$  of the second image.

Pseudo-code of iCSNs can be found in the Supplementary Materials.

**Training objective.** iCSNs are finally trained with a single pixel-wise reconstruction loss per paired image over batches of size  $N$ :

$$L = \frac{1}{2N} \sum_{i=1}^N (x_i - \hat{x}_i)^2 + (x'_i - \hat{x}'_i)^2 \quad (3)$$

This simple loss term stands in contrast to several previous works on prototype learning, which enforce semantic binding via an additional consistency loss [43, 62, 55]. By including the semantic binding implicitly into the network architecture, we eliminate the need for additional hyperparameters and a more complex optimization process over multiple objectives.

**Interacting with iCSNs.** The goal of iCSNs, especially in comparison to VAEs, is not necessarily to be a generative latent-variable model that learns the underlying data distribution, but to learn prototypical concept representations that are human-understandable and interactable. The autoencoder structure is thus a means to an end rather than a necessity. However, instead of discarding the decoder after convergence, an iCSN can present an input sample’s closest prototypical reconstruction of each concept. Thus, by querying these prototypical reconstructions at test time, a human user can confirm whether the predicted concepts make sense and possibly detect undesired model behavior. By defining a threshold on the test time reconstruction error, an iCSN can give a heuristic indication of its certainty in recognizing concepts in novel samples.

Due to the discrete and semantically bound latent code  $y$ , a human user can easily interact with iCSNs by treating  $y$  as a multi-label one-hot encoding. Specifically, a human user can revise and add knowledge via additional loss terms to the full extent of Stammer *et al.* [75]. For example, via logical statements such as  $\forall \text{img.} \Rightarrow \neg \text{hasconcept}(\text{img}, p_1^1)$  or  $\forall \text{img. isin}(\text{img}, \text{imgset}) \Rightarrow \text{hasconcept}(\text{img}, p_2^1)$ , a user can formulate logical constraints which read as “Never detect the concept represented by the prototype  $p_1^1$ .” and “For every image in this set of images you should be detecting the concept represented by prototype  $p_2^1$ .”, respectively. The set of incorrectly represented images can be curated by the user interactively.

Lastly, the modularity of iCSNs has additional advantages or interactive online learning, *e.g.*, when the model is provided

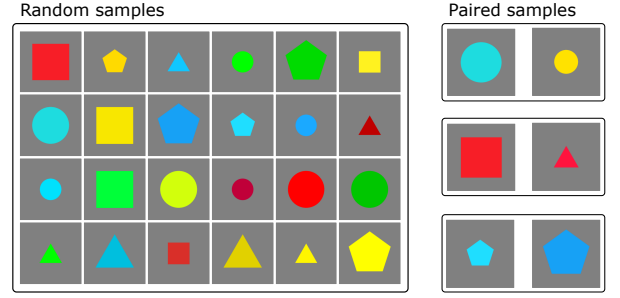


Figure 4: Samples of the Elementary Concept Reasoning data set. Each sample image (left) depicts a centered 2D object with three different properties: color, shape, and size. Images are paired such that the objects share between one and two concepts (right).

with data samples that contain novel concepts or when a factor that is present in the data is initially deemed unimportant but considered important retrospective to the initial learning phase. The approach for interaction in both cases depends on the hierarchy of the concept to be learned, namely if it is a basic concept or a superordinate concept. In the case of a novel basic concept, the approach is straightforward. Assuming human user satisfaction with the previous concept representations of an iCSN, and that  $J$ , the total number of prototype slots per codebook, was set to be overestimated, a user can simply give the feedback to represent a novel basic concept via one of the unused prototype slots of the relevant category.

In case a novel superordinate concept should be learned, one can apply a simple trick during the initial training phase by adding an additional read-out encoder  $\tilde{m}_{J+1}(z) = \tilde{\phi}_{J+1} \in \mathbb{R}^L$ . In contrast to the other read-out encoders, this one does not map to the space of the prototype slots where it would consecutively be discretized via  $S_{dot}(\cdot, \cdot)$  and  $S_\tau(\cdot)$ . Instead,  $\tilde{\phi}_{J+1}$  remains a continuous representation that is directly concatenated to the final latent codes  $y := [\Pi_0, \Pi_1, \dots, \Pi_J, \tilde{\phi}_{J+1}]$ . In this way,  $\tilde{m}_{J+1}(\cdot)$  can learn to incorporate all information that should not be discretized via the usual procedures. Ultimately, the initial latent space  $z$  of an iCSN can be trained to represent the full data distribution, even though only specific concepts should be discretized from this space. To include concepts that were initially considered irrelevant, one can just expand  $J$  which means to add a new read-out encoder  $m_{J+1}(z) = \phi_{J+1} \in \mathbb{R}^Q$  and codebook  $P_{J+1}$  to the iCSN. Then,  $m_{J+1}$  learns to bind novel basic concepts from the “novel” superordinate concept to  $P_{J+1}$  which only requires novel data pairs exemplifying the previously unimportant concept.

**Additional remarks.** To summarize: the gradient of 3 provides the learning signal for the entire network, including the initial encoder, read-out encoders, prototype slots and decoder. Furthermore, by decreasing the softmax temperature  $\tau$  in a step-wise fashion, one can enforce the binding of information to specific prototypes such that a specific concept is mapped to an individual prototype slot. Through this discretization process, the decoder learns to produce reconstructions that correspond to the prototypical representations present in the data. For example, given a pair of images of blue objects that vary in their shade of blue, an iCSN would learn to map the color



information of these objects to the same prototype slot, thus learning the prototypical blue of both shades. This discretization step is a key difference to the various GAN, and VAE approaches with Gaussian distributions, which try to learn a continuous latent space of the underlying factors.

## 4 Elementary Concept Reasoning Data Set

Recent studies show the benefits of object-centric learning for performing complex downstream tasks [83, 50, 75]. Thus, rather than learning concepts of an entire image, e.g. as Kim *et al.* [41], we introduce a novel benchmark data set, Elementary Concept Reasoning (ECR), which explicitly focuses on object-centric visual concept learning.

ECR consists of RGB images ( $64 \times 64 \times 3$ ) of 2D geometric objects on a constant colored background. Objects can vary in shape (circle, triangle, square, and pentagon), size (large and small), and color (red, green, blue, yellow). We add uniform jitter to each color, resulting in various color shades. Each image contains a single object which is fixed to the center of the image. Furthermore, ECR contains image pairs following the match pairing setup of Shu *et al.* [72]. We pair images such that the objects in the individual images share at least one, but at most  $J - 1$  common properties. ECR contains a training set size of 5000 image pairs and 2000 images for validation.

4 shows example images of ECR. Random samples are presented on the left, exemplifying shape, size, and color combinations. Example image pairs are presented on the right. An important feature of ECR is that although various shades of colors exist, they all map to four discrete colors. Notice, e.g., the color difference of the two paired blue objects. Even though both objects present different shades of blue, their state of being paired indicates that they share the same distinct shape (pentagon) and color (blue) concept.

## 5 Results

In this section, we demonstrate the advantages of prototype-based representations via Interactive Concept Swapping Networks. We begin our analysis by investigating the sparsity and semantics of iCSNs’ latent representations. Next, we show that the model can communicate the extracted concepts to a human user due to its discretized latent space. Subsequently, we simulate human user interactions via simple feedback rules, which are sufficient to revise an iCSNs’ latent concept space. Lastly, we show that novel concepts can easily be added into the concept space of iCSNs via simple human interactions.

**Experimental details.** For our experiments, we compare the iCSN to several baselines including the unsupervisedly-trained  $\beta$ -VAE [29] and Ada-VAE by Locatello *et al.* [49], using the arithmetic mean of the encoder distributions as in [33]. For a fair comparison with iCSNs which are trained via the shared match pairing of [72] and the Ada-VAE, which was originally introduced as a weaker form of supervision, we also trained the Ada-VAE with known shared factor IDs. This baseline essentially resembles a  $\beta$ -VAE with an averaging of encoder dis-

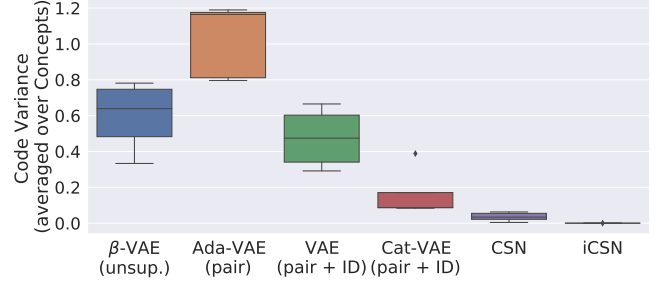


Figure 5: Average latent code variance given the ground truth concept labels for different model types and training settings. The compared models: unsupervisedly trained  $\beta$ -VAE, Ada-VAE with paired images, VAE and categorical VAE with paired images and known shared factor IDs, the novel CSN and iCSN with additional interactions on the learned concept space. Note that a lower variance is desirable.

tributions between pairs of images at the known shared factor IDs. It is denoted as VAE in the results below. Lastly, we compare to a discretizing VAE approach which uses a categorical distribution via the Gumbel-softmax trick [38, 51] (*Cat-VAE*). Cat-VAE is trained the same way as the VAE, i.e., via share pairing and averaging over encoder distributions.

We train the iCSN with a simple reconstruction loss as in 3 and decreasing softmax temperature, and present the results of two iCSN configurations. The vanilla setting, denoted as *Concept Swapping Network (CSN)*, corresponds to an iCSN prior to human interactions and contains an overestimated number  $K$  of prototype slots per superordinate concept. Although ECR contains a total number of four colors and shapes and two sizes, the vanilla CSN setting uses  $K = 6$  for each category. In our experiments  $J = 3$  corresponds to the ground truth number of superordinate concepts. Finally, *iCSN* denotes the case with additional user interactions with the vanilla CSN after the initial training phase.

The number of latent variables for  $\beta$ -VAE, Ada-VAE, and VAE was set to the ground truth number of superordinate concepts. All Cat-VAE runs were performed with three categorical distributions each with  $k = 6$  events.

All configurations were trained with five random seed initializations, and the results present the mean and standard deviation of these runs. All presented results were obtained from a held-out validation set. Additional hyperparameters, architectural details, and detailed descriptions of evaluations metrics and simulated user interactions can be found in the Supplementary Materials.<sup>1</sup>

**Reduced code variance.** Human understandability of and interactions with a model’s latent space strongly benefit from consistency in a model’s concept representations. In other words, the representation of the color blue should be specific to this concept and the latent representation for the blue color of one object should be very similar to that of a second blue object. If this is not the case, it remains difficult for a human user to identify and interact with these learned concepts.

Motivated by this intuition, we first investigate the variance of the latent representations given the ground truth multi-label information of each validation image. For this, we compute the

<sup>1</sup>Code can be found on [github after acceptance](#)

	DT	LR
$\beta$ -VAE (unsup.)	$88.98 \pm 11.53$	$73.86 \pm 18.07$
Ada-VAE (pair)	$63.39 \pm 11.32$	$74.07 \pm 6.40$
VAE (pair + ID)	$92.18 \pm 6.26$	$97.80 \pm 1.09$
Cat-VAE (pair + ID)	$74.79 \pm 19.45$	$91.17 \pm 13.13$
<b>CSN</b>	<b><math>99.92 \pm 0.05</math></b>	<b><math>99.87 \pm 0.07</math></b>
<b>iCSN</b>	<b><math>100.0 \pm 0.00</math></b>	<b><math>100.0 \pm 0.00</math></b>

Table 1: Linear probing via decision tree (DT) and logistic regression (LR) on the latent codes of different model types. The classification accuracies were computed on a held out test set.

latent code variance over all validation images of each concept. In mathematical notation this corresponds to:

$$\frac{1}{K \cdot J} \sum_{j=1}^{J=3} \sum_{k=1}^K \text{Var}(\{\bar{z}_j\}_{l=k}) \quad (4)$$

with  $\text{Var}(\cdot)$  denoting the variance,  $\bar{z}_j$  denoting a place holder for the latent representation of a corresponding model (the discretized prototype distance code  $y$  for iCSNs, the distribution means for VAEs with Gaussian distributions and the event probabilities for Cat-VAE).  $\{\cdot\}_{l=k}$  denotes the set of latent representations from images for which the ground truth basic concept of category  $j$  corresponds to  $k$ .

The resulting code variances over all models can be seen in 5. Note that a low code variance is desirable and indicates how well a concept is mapped to a distinct representation. The results in 5 suggest that the variance of the latent space from CSNs is much lower, showing more consistent concept representations. However, a reduced latent code variance is not a sufficient criterion for concept consistency and human understandability. For example, a model that learns to map all concepts to a single representation has zero latent code variance but also no representational power. Therefore, we turn to probing the latent concept space via linear models next.

**Probing the latent space.** Similar to works of the self-supervision community [10, 61, 14, 12, 15], we investigate the latent code of each model via linear probing. For this, the latent codes of each model on a held-out data set are inferred, as in the previous experiment. Next, ground truth labels are obtained by converting each multi-label ground truth vector,  $c$ , of this data set to a 32-dimensional one-hot encoding. Finally, a Decision Tree (DT) and a Logistic Regression (LR) are trained supervisedly on this data set and validated on an additional held-out data set.

The results in 1 document the average accuracy and standard deviation on the held-out validation set over the five random initializations for the different models. We observe that the latent code of CSNs allows for nearly perfect predictive performance and surpasses all variational approaches. Importantly, CSNs’ representations even surpass those of VAE approaches (VAE and Cat-VAE) that were trained with the same type of weak supervision as CSNs. As expected, the  $\beta$ -VAE performs worse on average than the weakly-supervised models. Interestingly, however, the Ada-VAE configuration performed worse than the  $\beta$ -VAE. In addition, the discrete latent representations of Cat-VAE also perform worse than CSNs. Noticeably, the Cat-VAE runs indicate a high deviation in per-

formance, indicating that several Cat-VAE runs converged to sub-optimal states. In summary, although the ECR data set only contains variations in individual 2D geometrical objects, the baseline models do not perform as well as CSNs, even when trained with the same amount of information.

**Explaining and revising the latent space.** An advantage of an iCSN’s semantically bound, discrete latent space, is the straightforward identification of sub-optimal concept representations by a human user *cf.* 1. Upon identifying correctly or falsely learned concepts, a user can then apply simple logical feedback rules on this discrete concept space.

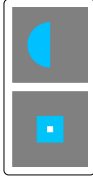
Specifically, after training via weak supervision, it is recommendable for the machine and human user to discuss the learned concepts and identify whether these coincide with the user’s knowledge or if a revision is necessary. For example, an iCSN can learn to represent a color over several prototype slots or represent two shapes via one slot, indicating that it falsely considers these to belong to the same concept. An iCSN can then convey its learned concepts in two ways. First, it can group novel images that share a concept according to its inferred discrete prototype distance codes and inquire a human user if indeed the grouped images share a common underlying concept *cf.* 1. Second, utilizing the decoder, it can present the prototypical reconstruction of each learned concept, *e.g.*, presenting an object with a prototypical shade of blue *cf.* 2.

Having identified potential sub-optimal concept representations, a human user can now interact on the discretized latent space of iCSNs via logical rules and further improve the representations, which we demonstrate via simulated user interactions in the following. For all previous runs of the vanilla CSN configuration, we visually inspect the concept encodings  $y$  for one example each of the 32 possible concept combinations and identify those prototype slots which are “activated” in the majority of examples per individual concept (*primary* slots) and, additionally, identify those prototype slots per concept that are never or rarely activated within our subset of examples (*secondary* slots). We next apply an  $L2$  loss on  $y$  to never use these *secondary* slots and finetune the previous runs on the original training set with the original reconstruction loss and this additional  $L2$  loss. The semantics of this feedback is that concepts should only be represented by their *primary* prototype slots. Additionally, in two runs we revise an observed sub-optimal solution that pentagons and circles are bound to the same prototype slot. Hereby, feedback is provided on all pentagon samples of the training set to bind to an otherwise empty prototype slot, again via an additional  $L2$  loss.

The results of these simple interactions can be found under iCSN in 5 and 1 indicating a near-zero latent code variance per ground truth concept and perfect linear probing accuracy, respectively. Thus, indicating the ease of interacting with and revising the latent space of iCSNs.

**Interactively learning novel basic concept.** Furthermore, the prototype-based representations of iCSNs possess interesting properties for an online learning setting, *e.g.*, when encountering novel concepts which the model has not seen before. Through the decoder of the iCSN, evaluating the reconstruction and reconstruction error can serve as a means for identifying whether the model has a good representation of a novel sample *cf.* 2.

When teaching an iCSN a novel basic concept like a *halfcir-*



	iCSN	DT	LR
Novel Basic Concept			
before		78.87 $\pm$ 1.31	78.73 $\pm$ 1.22
after		<b>98.3 <math>\pm</math> 1.32</b>	<b>98.28 <math>\pm</math> 1.33</b>
Novel Superordinate Concept			
before		93.1 $\pm$ 4.46	67.54 $\pm$ 9.07
after		<b>99.85 <math>\pm</math> 0.3</b>	<b>98.29 <math>\pm</math> 3.42</b>

Figure 6: Linear probing via Decision Tree (DT) and Logistic Regression (LR) on the latent codes of iCSN. We evaluate the models with a ECR data set containing a novel shape (top) and a novel superordinate spot concept (bottom), each not seen during initial training (before). After human user interactions (after) this novel information could be easily added to the concept representations.

*cle* shape *cf.* 6, a user can identify an unbound prototype slot of the model’s latent representation and encourage the binding to this slot. To prevent catastrophic forgetting [53, 63], i.e., overriding already learned concepts, we employ a combination of common rehearsal [65] and knowledge distillation methods [30, 45] by letting the model predict the latent codes of past samples and restrict the iCSN to not deviate from those. Specifically, we use a simple  $L2$  loss on the known and unknown one-hot concept encodings to encourage binding of the unknown concepts while not forgetting the known ones.

6 (top) shows the linear model prediction accuracies on the latent space of iCSNs that have been presented a data set containing the novel concept *halfcircle*. The tag *before* indicates the accuracy on the latent code of the iCSN runs of 1 that were trained on the standard ECR concepts, whereas *after* indicates the accuracy on the latent code after additional interactions with a simulated user by providing the information which empty prototype slot of the shape codebook to bind the novel concept to. These results indicate the ease of adding additional knowledge on novel basic concepts to the latent representation of iCSNs.

**Interactively learning novel superordinate concept.** Next, we showcase how to add a novel superordinate concept. For this setting, we make use of a variation of the ECR data set where white spots were added to the center of roughly half of the objects *cf.* 6. The other half depicts a solid color as in the original ECR data set.

We consider an online learning setting where spots are unimportant during the initial training but reconsidered important in the second round of interaction. The modularity of iCSNs allows us to easily add a read-out encoder during initial training that learns to represent all the information of the data that is not discretized via the initial paired samples (*cf.* 3 for details). In the second round of training, this continuous read-out encoder can be replaced with a discretizing read-out encoder and additional codebook. This modularity property further eliminates the danger of catastrophic forgetting in that all previously trained modules can be frozen in the second training round, thus only requiring the novel read-out encoder to be finetuned.

Human user interactions are simulated by assuming that the iCSN has correctly learned the previous concepts with an additional continuous read-out encoder for representing the spotted

	DT	LR
Cat-VAE w. swap.	60.85 $\pm$ 7.29	65.71 $\pm$ 10.66
iCSN w. avg.	68.84 $\pm$ 39.4	68.81 $\pm$ 39.4

Table 2: Linear probing via Decision Tree (DT) and Logistic Regression (LR) on the latent codes of Cat-VAE with encoder distribution swapping and iCSN concept encoding averaging.

feature. Subsequently, additional training pairs are introduced that exemplify the novel superordinate concept, and the new read-out encoder is finetuned as in the standard training setting. Results can be seen in 6 again presenting the linear model accuracy on the latent representations before and after simulated user interactions, indicating that the novel superordinate concept can easily be bound to the model’s internal prototype representations.

Here, we remark on desirable properties of iCSNs for handling confounded data. Assuming an undesired confounding factor within the data generation process that causes spurious features, an iCSN can learn to ignore these features during its training process via the mechanism presented above. This stands in comparison to GANs or variational approaches with Gaussian distributions, which could potentially learn also to model the spurious features.

**Ablation studies.** To assess the importance of the different components of iCSNs, we conduct an ablation study and depict the linear probing classification performances in 2. Specifically, we test a Cat-VAE that uses swapping of the relevant encoder distributions, rather than averaging as in previous experiments. And, secondly, we test an iCSN with averaging of the concept encodings  $\phi_j$ , rather than swapping. With these experiments, we wish to (i) compare the discretization via distances to prototypes to discretization via categorical distributions (Cat-VAE w. swapping) and (ii) test the influence of swapping versus averaging of encodings for iCSNs (iCSN w. avg).

The results of 2 when compared with 1 indicate that discrete representations via distances to prototypes are, in fact, beneficial compared to those of the categorical distributions of Cat-VAEs. Secondly, the swapping procedure appears to be crucial for optimal learning of concept representations in iCSNs.

**Limitations.** Following assumptions were made in this work: a superordinate concept is divisible into multiple basic concepts and “valid” user feedback was provided in our experiments on interactions. A potential limiting factor of iCSNs is the training reconstruction loss which might be insufficient for learning fine-grained concepts. Additionally, we observed that the choice of  $\tau$  can influence the quality of the learning process. Setting a small value too early can lead to sub-optimal solutions. Lastly, our approach was tested on ECR due to its object-centric nature and distinct concept distribution of the data set. For more complex settings additional architectures may be required to pre-process the data to *e.g.* extract objects from an image.

## 6 Conclusion

In this work, we investigated the properties of latent prototype representations for neural concept learning with weak supervi-



sion. The results with our novel iCSN framework indicate that these are beneficial for human-understandable concept learning but also human interactions and the incorporation of novel concepts within an online learning setting. Interesting pathways for future research are applying iCSNs to more complex data sets, particularly from critical domains such as medical or scientific data where often relevant concepts are not known in advance, however standard deep learning approaches can learn to focus on confounding factors, *e.g.* for Covid-19 data [18] or plant phenotyping data [68]. We hypothesize the interactive approach of iCSNs to be beneficial in allowing the machine and human to identify relevant and irrelevant concepts within the data jointly.

## 7 Acknowledgements

This work has been supported by the BMEL/BLE under the innovation support program, project “AuDiSens” (FKZ28151NA187). Additionally, it has benefited from the HMWK cluster project “The Third Wave of AI”. The authors wish to thank Cigdem Turan for providing figure sketches and Martin Mundt for useful feedback during the writing process.

## References

- [1] David Alvarez-Melis and Tommi S. Jaakkola. Towards robust interpretability with self-explaining neural networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018. 2
- [2] E James Archer. The psychological nature of concepts. In *Analyses of concept learning*, pages 37–49. Elsevier, 1966. 2
- [3] Catarina Belém, Vladimir Balayan, Pedro Saleiro, and Pedro Bizarro. Weakly supervised multi-task learning for concept-based explainability. *arXiv preprint arXiv:2104.12459*, 2021. 2
- [4] Yoshua Bengio, Aaron C. Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2013. 2
- [5] Diane Bouchacourt, Ryota Tomioka, and Sebastian Nowozin. Multi-level variational autoencoder: Learning disentangled representations from grouped observations. *AAAI Conference on Artificial Intelligence*, 2018. 2, 4
- [6] Caitlin R Bowman, Takako Iwashita, and Dagmar Zeithamova. Tracking prototype and exemplar representations in the brain across learning. *eLife*, 2009. 2
- [7] Christopher P. Burgess, Loïc Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matthew Botvinick, and Alexander Lerchner. Monet: Unsupervised scene decomposition and representation. *Computing Research Repository (CoRR)*, 2019. 3
- [8] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision (ECCV)*, 2020. 3
- [9] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 4
- [10] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 6
- [11] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: Deep learning for interpretable image recognition. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019. 2
- [12] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, 2020. 6
- [13] Xi Chen, Yan Duan, Rein Houthoofd, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2016. 1
- [14] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 6
- [15] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. *arXiv preprint arXiv:2104.02057*, 2021. 6
- [16] Gabriele Ciravegna, Pietro Barbiero, Francesco Giannini, Marco Gori, Pietro Lió, Marco Maggini, and Stefano Melacci. Logic explained networks. *arXiv preprint arXiv:2108.05149*, 2021. 2
- [17] Stanley J. Colcombe and Robert S. Wyer. The role of prototypes in the mental representation of temporally related events. *Cognitive Psychology*, pages 67–103, 2002. 2
- [18] Alex J DeGrave, Joseph D Janizek, and Su-In Lee. Ai for radiographic covid-19 detection selects shortcuts over signal. *Nature Machine Intelligence*, 2021. 1, 8
- [19] Martin Engelcke, Adam R. Kosiorek, Oivi Parker Jones, and Ingmar Posner. GENESIS: generative scene inference and sampling with object-centric latent representations. In *ICLR*, 2020. 3
- [20] Michael W Eysenck and Marc Brysbaert. *Fundamentals of cognition*. Routledge, 2018. 3
- [21] Jerry A Fodor. *Concepts: Where cognitive science went wrong*. Oxford University Press, 1998. 2
- [22] Marcello Frixione. and Antonio Lieto. Prototypes vs exemplars in concept representation. In *International Conference on Knowledge Engineering and Ontology Development (KEOD)*, 2012. 2
- [23] Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. Towards automatic concept-based explanations. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019. 2
- [24] Ross B. Girshick. Fast R-CNN. In *International Conference on Computer Vision (ICCV)*, 2015. 3
- [25] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2014. 1
- [26] Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Christopher Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-object representation learning with iterative variational inference. In *International Conference on Machine Learning (ICML)*, 2019. 3
- [27] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *International Conference on Computer Vision (ICCV)*, 2017. 3
- [28] William Heindel, Elena Festa, Brian Ott, Kelly Landy, and David Salmon. Prototype learning and dissociable categorization systems in alzheimer’s disease. *Neuropsychologia*, 51, 2013. 2



- [29] Irina Higgins, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations (ICLR)*, 2017. 1, 5
- [30] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 7
- [31] Haruo Hosoya. Group-based learning of disentangled representations with generalizability for novel contents. *arXiv preprint arXiv:1809.02383*, 2018. 2
- [32] Haruo Hosoya. Group-based learning of disentangled representations with generalizability for novel contents. In *Joint Conference on Artificial Intelligence (IJCAI)*, 2019. 2
- [33] Haruo Hosoya. Group-based learning of disentangled representations with generalizability for novel contents. In Sarit Kraus, editor, *Joint Conference on Artificial Intelligence (IJCAI)*, 2019. 4, 5
- [34] Yordan Hristov, Alex Lascarides, and Subramanian Ramamoorthy. Interpretable latent spaces for learning from demonstration. In *Conference on Robot Learning (CoRL)*, 2018. 2
- [35] Shixin Huang, Xiangping Zeng, Si Wu, Zhiwen Yu, Mohamed Azzam, and Hau-San Wong. Behavior regularized prototypical networks for semi-supervised few-shot image classification. *Pattern Recognition*, 2021. 2
- [36] Paul Ibbotson and Michael Tomasello. Prototype constructions in early language acquisition. *Cambridge University Press*, pages 59–85, 2009. 2
- [37] Frank Jäkel, Bernhard Schölkopf, and Felix A. Wichmann. Generalization and similarity in exemplar models of categorization: Insights from machine learning. *Psychonomic Bulletin & Review*, 2008. 2
- [38] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations (ICLR)*, 2017. 5
- [39] Insu Jeon, Wonkwang Lee, Myeongjae Pyeon, and Gunhee Kim. Ib-gan: Disentangled representation learning with information bottleneck generative adversarial networks. In *AAAI Conference on Artificial Intelligence*, 2021. 1
- [40] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *Proceedings of the 35th International Conference on Machine Learning*, Proceedings of Machine Learning Research (PMLR), 2018. 2
- [41] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International Conference on Machine Learning (ICML)*, 2018. 5
- [42] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *Proceedings of Machine Learning Research (PMLR)*, 2020. 1, 2
- [43] Oscar Li, Hao Liu, Chaofan Chen, and Cynthia Rudin. Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In *AAAI Conference on Artificial Intelligence*, 2018. 2, 4
- [44] Xiaoqiang Li, Liangbo Chen, Lu Wang, Pin Wu, and Weiqin Tong. Segan: Disentangled representation learning by adding similarity constraint on generative adversarial nets. *IEEE Access*, 2019. 1
- [45] Zhizhong Li and Derek Hoiem. Learning without forgetting. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *European Conference on Computer Vision (ECCV)*, 2016. 7
- [46] Zhixuan Lin, Yi-Fu Wu, Skand Vishwanath Peri, Weihao Sun, Gautam Singh, Fei Deng, Jindong Jiang, and Sungjin Ahn. SPACE: unsupervised object-oriented scene representation via spatial attention and decomposition. In *International Conference on Learning Representations (ICLR)*, 2020. 3
- [47] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning (ICML)*, 2019. 1, 2
- [48] 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 *International Conference on Machine Learning (ICML)*, 2019. 3
- [49] Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In *International Conference on Machine Learning (ICML)*, 2020. 1, 4, 5, 11
- [50] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 3, 5
- [51] Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *International Conference on Learning Representations (ICLR)*, 2017. 5
- [52] Diego Marcos, Ruth Fong, Sylvain Lobry, Rémi Flamary, Nicolas Courty, and Devis Tuia. Contextual semantic interpretability. In Hiroshi Ishikawa, Cheng-Lin Liu, Tomas Pajdla, and Jianbo Shi, editors, *Asia Conference on Computer Vision (ACCV)*, 2021. 2
- [53] Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Academic Press, 1989. 7
- [54] Marius Memmel, Camila Gonzalez, and Anirban Mukhopadhyay. Adversarial continual learning for multi-domain hippocampal segmentation. In *Domain Adaptation and Representation Transfer (DART) at International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2021. 1
- [55] Umberto Michieli and Mete Ozay. Prototype guided federated learning of visual feature representations. *arXiv preprint arXiv:2105.08982*, 2021. 4
- [56] Graziano Mita, Maurizio Filippone, and Pietro Michiardi. An identifiable double vae for disentangled representations. In *International Conference on Machine Learning (ICML)*, 2021. 1
- [57] Thu H Nguyen-Phuoc, Christian Richardt, Long Mai, Yongliang Yang, and Niloy Mitra. Blockgan: Learning 3d object-aware scene representations from unlabelled images. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 1
- [58] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1
- [59] Daniel M. Oppenheimer, Joshua B. Tenenbaum, and Tevye R. Kyrnski. Chapter six - categorization as causal explanation: Discounting and augmenting in a bayesian framework. In *Categorization as causal reasoning*, Psychology of Learning and Motivation, pages 203–231. Academic Press, 2013. 2
- [60] Frederik Pahde, Mihai Puscas, Tassilo Klein, and Moin Nabi. Multimodal prototypical networks for few-shot learning. In

- IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021. [2](#)
- [61] Massimiliano Patacchiola and Amos Storkey. Self-supervised relational reasoning for representation learning. *arXiv preprint arXiv:2006.05849*, 2020. [6](#)
- [62] Mihir Prabhudesai, Shamit Lal, Darshan Patil, Hsiao-Yu Tung, Adam W Harley, and Katerina Fragkiadaki. Disentangling 3d prototypical networks for few-shot concept learning. In *International Conference on Learning Representations (ICLR)*, 2021. [2, 4](#)
- [63] Roger Ratcliff. Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. *Psychological review*, 97(2):285, 1990. [7](#)
- [64] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. [3](#)
- [65] A. Robins. Catastrophic forgetting in neural networks: the role of rehearsal mechanisms. In *Proceedings 1993 The First New Zealand International Two-Stream Conference on Artificial Neural Networks and Expert Systems*, pages 65–68, 1993. [7](#)
- [66] Andrew Slavin Ross, Nina Chen, Elisa Zhao Hang, Elena L. Glassman, and Finale Doshi-Velez. Evaluating the interpretability of generative models by interactive reconstruction. In *Conference on Human Factors in Computing System (CHI)*, 2021. [1, 2](#)
- [67] Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations. In *Joint Conference on Artificial Intelligence (IJCAI)*, 2017. [2](#)
- [68] Patrick Schramowski, Wolfgang Stammer, Stefano Teso, Anna Brugger, Franziska Herbert, Xiaoting Shao, Hans-Georg Luigs, Anne-Katrin Mahlein, and Kristian Kersting. Making deep neural networks right for the right scientific reasons by interacting with their explanations. *Nature Machine Intelligence*, 2020. [1, 2, 8](#)
- [69] Norbert M. Seel, editor. *Prototype*, pages 2714–2714. Springer US, 2012. [2](#)
- [70] Ramprasaath Ramasamy Selvaraju, Stefan Lee, Yilin Shen, Hongxia Jin, Shalini Ghosh, Larry P. Heck, Dhruv Batra, and Devi Parikh. Taking a HINT: leveraging explanations to make vision and language models more grounded. In *International Conference on Computer Vision (ICCV)*, 2019. [2](#)
- [71] Andrew W. Senior, Richard Evans, John Jumper, James Kirkpatrick, Laurent Sifre, Tim Green, Chongli Qin, Augustin Zidek, Alexander W. R. Nelson, Alex Bridgland, Hugo Penedones, Stig Petersen, Karen Simonyan, Steve Crossan, Pushmeet Kohli, David T. Jones, David Silver, Koray Kavukcuoglu, and Demis Hassabis. Improved protein structure prediction using potentials from deep learning. *Nat.*, 577(7792):706–710, 2020. [1](#)
- [72] Rui Shu, Yining Chen, Abhishek Kumar, Stefano Ermon, and Ben Poole. Weakly supervised disentanglement with guarantees. In *International Conference on Learning Representations (ICLR)*, 2020. [1, 2, 3, 5](#)
- [73] J. David Smith. Prototypes, exemplars, and the natural history of categorization. *Psychonomic bulletin & review*, 2014. [2](#)
- [74] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2017. [2](#)
- [75] Wolfgang Stammer, Patrick Schramowski, and Kristian Kersting. Right for the right concept: Revising neuro-symbolic concepts by interacting with their explanations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. [2, 4, 5](#)
- [76] J.R. Taylor. *Linguistic Categorization*. OUP Oxford, 2003. [2](#)
- [77] Stefano Teso and Kristian Kersting. Explanatory interactive machine learning. In *AAAI/ACM Conference on AI, Ethics, and Society (AIES)*, 2019. [2](#)
- [78] Michael Tschannen, Olivier Bachem, and Mario Lucic. Recent advances in autoencoder-based representation learning. *arXiv preprint arXiv:1812.05069*, 2018. [1](#)
- [79] Matthew J. Vowels, Necati Cihan Camgoz, and Richard Bowden. Gated variational autoencoders: Incorporating weak supervision to encourage disentanglement. In *IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, 2020. [1](#)
- [80] Yuxin Wu and Kaiming He. Group normalization. In *European Conference on Computer Vision (ECCV)*, 2018. [11](#)
- [81] 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. [2](#)
- [82] Chih-Kuan Yeh, Been Kim, Sercan Arik, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. On completeness-aware concept-based explanations in deep neural networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020. [2](#)
- [83] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. Neural-symbolic VQA: disentangling reasoning from vision and language understanding. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018. [1, 2, 5](#)
- [84] Julian Zaidi, Jonathan Boilard, Ghyslain Gagnon, and Marc-André Carbonneau. Measuring disentanglement: A review of metrics. *arXiv preprint arXiv:2012.09276*, 2020. [2](#)
- [85] Dagmar Zeithamova. *Prototype Learning Systems*, pages 2715–2718. Springer US, 2012. [2](#)
- [86] Xinqi Zhu, Chang Xu, and Dacheng Tao. Where and what? examining interpretable disentangled representations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. [2](#)

# Interactive Disentanglement: Learning Concepts by Interacting with their Prototype Representations

## Supplementary Materials

### Hyperparameters and model details

#### Interactive Concept Swapping Networks

In our experiments, the prototype slots were initialized randomly from a truncated Gaussian distribution with mean  $\mu = 0$ , variance  $\sigma^2 = 0.5$ , minimum  $a = -1$ , and maximum  $b = 1$ . The encoder used in our experiments was a convolutional neural network with residual connections and ReLU activations. Each read-out encoder is a linear layer with LeakyReLU activations. Lastly, the decoder architecture was again a neural network with transposed convolutions and also here residual layers.

In the standard ECR experiments with iCSNs,  $J = 3$ ,  $Z = 512$ ,  $Q = 128$ ,  $K = 6$  for each  $j \in [1, \dots, J]$ ,  $N = 128$  and  $\tau$  was decreased every 1000 epochs with steps  $[2., 0.5, 0.1, 0.01, 0.001, 0.0001, 0.00001]$  over 8000 epochs in total.

Notably, group normalization as proposed by Wu *et al.* [80] was applied after extracting the concept encodings via the read-out encoders (performed in `collectedReadOutEncoders` in Alg. 1).

Pseudo code can be found in Alg. 1 and Alg. 2.

#### Baseline models

For the experiments with Cat-VAE, the softmax temperature was set to  $\tau = 0.1$  and each categorical distribution had  $k = 6$  categories. The number of latent variables was set to 3 for Cat-VAE,  $\beta$ -VAE, Ada-VAE, and VAE runs. For all baselines, the encoder and decoder consisted of a convolutional and transposed convolutional network. In all experiments,  $\beta = 4.$ , except for Ada-VAE, where  $\beta = 1.$ , as recommended by Locatello *et al.* [49]. All baseline models were trained for 2000 epochs.

#### Linear probing

The linear models for probing the latent representations of the different model configurations were a decision tree and logistic regression model. The max depth of the decision tree was set to 8. The logistic regression model was run with parameters  $C = 0.316$  and the maximum number of iterations at 1000. Both the decision tree and logistic regression model were trained with a fixed random seed.

#### Details on simulated interactions

The simulated user interactions were performed via an  $L_2$  regulatory loss term on the latent codes  $y$ .

In case a user tells an iCSN *not* to use a specific prototype slot of the superordinate concept  $j$  and slot  $k$ , this loss corresponds to:  $MSE(y_{j \cdot k + k}, \mathbf{0})$ , where  $y_{j \cdot k + k}$  corresponds to the value of  $y$  at position  $j \cdot k + k$  and  $\mathbf{0}$  being a vector of length  $N$ .

When a user provides a subset of examples with corresponding desired prototype slot IDs the loss term corresponds to:  $MSE(y_{j \cdot k + k}^{subset}, \mathbf{1})$  with  $\mathbf{1}$  of length equal to the number of samples in *subset*. The subset of examples in our simulated interactions were identified via the ground truth labels, e.g., for identifying the subset of images containing a pentagon. The interactions for learning a novel basic concept followed the same procedure.

To allow the model to update its latent space via interactions we increased  $\tau = 0.00001$  back to  $\tau = 0.0001$ .



---

**Algorithm 1:** Interactive Concept Swapping Network – pair images forward pass

---

**Input** : Image pair  $x \in \mathbb{R}^D, x' \in \mathbb{R}^D$ , known share IDs  $v$ .  
**Output:** Image reconstructions  $\hat{x} \in \mathbb{R}^D, \hat{x}' \in \mathbb{R}^D$ , and latent codes  $y \in [0, 1]^{J \cdot K}, y' \in [0, 1]^{J \cdot K}$

```
1 // Forward pass through initial encoder.  $z \in \mathbb{R}^Z$ 
2  $z \leftarrow f(x)$ 
3  $z' \leftarrow f(x')$ 
4 // Forward pass through  $J$  read-out encoders.  $\phi \in \mathbb{R}^{J \times Q}$ 
5  $\phi \leftarrow \text{collectedReadOutEncoders}(z)$ 
6  $\phi' \leftarrow \text{collectedReadOutEncoders}(z')$ 
7 // Compute the distance of each concept encoding to all prototype slots of its
  corresponding category  $j$ .
8  $y \leftarrow \text{computeProtoDistance}(\phi, v)$ 
9  $y' \leftarrow \text{computeProtoDistance}(\phi', v)$ 
10 // Reconstruct the images from the prototype distance codes.
11  $\hat{y} \leftarrow g(y)$ 
12  $\hat{y}' \leftarrow g(y')$ 
```

---

---

**Algorithm 2:** computeProtoDistance

---

**Input** : Concept encodings  $\phi \in \mathbb{R}^{J \times Q}, \phi' \in \mathbb{R}^{J \times Q}$   
**Given** : Set of prototype slot codebooks  $\Theta := [P_1, \dots, P_J] \in \mathbb{R}^{J \times Q \times K}$ , softmax temperature  $\tau$ , and share IDs  $v$ .  
**Output:** Latent codes  $y \in [0, 1]^{J \cdot K}, y' \in [0, 1]^{J \cdot K}$

```
1 // For every superordinate concept category
2 for  $j \leftarrow 0$  to  $J - 1$  do
3   // Dot-product between concept encoding and all prototype slots from codebook
    $P_j$ .
4    $s_j \leftarrow \text{softmaxDotProduct}(\phi_j, P_j)$ 
5    $s'_j \leftarrow \text{softmaxDotProduct}(\phi'_j, P_j)$ 
6   // Compute normalizing weighted softmax.
7    $\Pi_j \leftarrow \text{softmaxNormTau}(s_j, \tau)$ 
8    $\Pi'_j \leftarrow \text{softmaxNormTau}(s'_j, \tau)$ 
9 end for
10 // Swap the distance codes at the position corresponding to the shared IDs.
11  $y \leftarrow [\Pi_1, \dots, \Pi'_v, \dots, \Pi_J]$ 
12  $y' \leftarrow [\Pi'_1, \dots, \Pi_v, \dots, \Pi'_J]$ 
```

---