SCAN: Learning Abstract Hierarchical Compositional Visual Concepts

Irina Higgins, Nicolas Sonnerat, Loic Matthey, Arka Pal, Christopher Burgess, Matthew Botvinick, Demis Hassabis, Alexander Lerchner
Google DeepMind
{irinah,sonnerat,lmatthey,arkap,cpburgess,
botvinick,demishassabis,lerchner}@google.com

Abstract

The natural world is infinitely diverse, yet this diversity arises from a relatively small set of coherent properties and rules, such as the laws of physics or chemistry. We conjecture that biological intelligent systems are able to survive within their diverse environments by discovering the regularities that arise from these rules primarily through unsupervised experiences, and representing this knowledge as abstract concepts. Such representations possess useful properties of compositionality and hierarchical organisation, which allow intelligent agents to recombine a finite set of conceptual building blocks into an exponentially large set of useful new concepts. This paper describes SCAN (Symbol-Concept Association Network), a new framework for learning such concepts in the visual domain. We first use the previously published β-VAE (Higgins et al., 2017a) architecture to learn a disentangled representation of the latent structure of the visual world, before training SCAN to extract abstract concepts grounded in such disentangled visual primitives through fast symbol association. Our approach requires very few pairings between symbols and images and makes no assumptions about the choice of symbol representations. Once trained, SCAN is capable of multimodal bi-directional inference, generating a diverse set of image samples from symbolic descriptions and vice versa. It also allows for traversal and manipulation of the implicit hierarchy of compositional visual concepts through symbolic instructions and learnt logical recombination operations. Such manipulations enable SCAN to invent and learn novel visual concepts through recombination of the few learnt concepts.

1 Introduction

State of the art deep learning approaches to machine learning have achieved impressive results in many problem domains, including classification (He et al., 2016; Szegedy et al., 2015), density modelling (Gregor et al., 2015; Oord et al., 2016a,b), and reinforcement learning (Mnih et al., 2015, 2016; Jaderberg et al., 2017; Silver et al., 2016). They are still, however, far from possessing many traits characteristic of human intelligence. Such deep learning techniques tend to be overly data hungry, often rely on significant human supervision, tend to overfit to the training data distribution and are brittle to even small domain shifts (Lake et al., 2016; Garnelo et al., 2016). An important first step towards bridging the gap between human and artificial intelligence is endowing algorithms with compositional concepts (Lake et al., 2016; Garnelo et al., 2016). Compositionality allows for reuse of a finite set of primitives across many various scenarios by recombining them to produce an exponentially large number of novel yet coherent and useful concepts. Compositionality is at the core of such human abilities as creativity, imagination and language based communication. It is no surprise that the use of compositional concepts has, through evolution, become a hallmark of higher levels of intelligence, since it provides a way of dealing with the seemingly infinite diversity of the natural world by exploiting its relatively simple and coherent underlying structure (e.g. the ubiquitous regularities due to the laws of physics or chemistry).

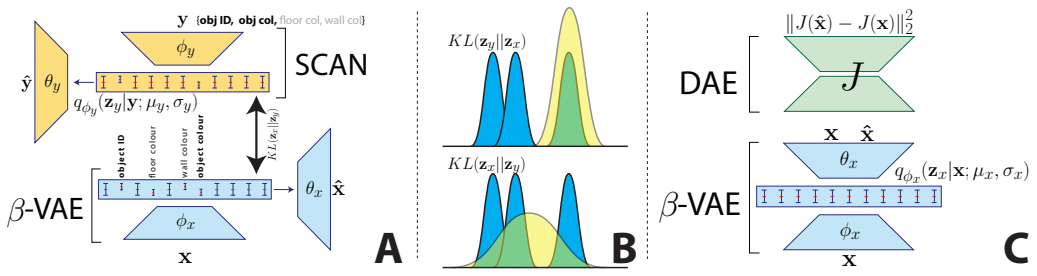

Figure 1: **A**: SCAN model architecture. **B**: Mode coverage of the extra KL term of the SCAN loss function. Each blue mode corresponds to an inferred visual latent distribution $q(z_x^k|x_i)$ given an image x_i. The yellow distribution corresponds to the learnt conceptual latent distribution $q(z_y^k)$. When presented with visual examples that have high variability for a particular generative factor, e.g. various lighting conditions when viewing examples of apples, it is important to learn a broad distribution for the corresponding conceptual latent $q(z_y^k)$ that is close to the prior distribution for the data generative factor. This can be achieved through forward KL divergence $D_{KL}(\mathbf{z}_x||\mathbf{z}_y)$, rather than the mode picking reverse KL divergence $D_{KL}(\mathbf{z}_y||\mathbf{z}_x)$. **C**: β-VAE$_{DAE}$ model architecture.

Intelligent agents, such as humans, are able to discover and learn abstract concepts in a largely unsupervised manner (e.g. intuitive physics (Lake et al., 2016; Smith & Vul, 2013; Baillargeon, 1987, 2004; Spelke, 1990)). These concepts tend to be grounded in the latent structure of the world (e.g. the concept of an apple is defined in terms of the latent object properties, such as colour, shape and size). This paper describes SCAN (Symbol-Concept Association Network), a neural network implementation of such a learning process (Fig. 1A). In particular, our model is able to learn grounded visual concepts in a largely unsupervised manner through fast symbol association.

We hypothesise that concepts are abstractions over a set of visual primitives. For example, consider a toy hierarchy of concepts shown in Fig. 2. Each node in this hierarchy is defined as a subset of four visual primitives that describe the scene in the input image: object identity (I), object colour (O), floor colour (F) and wall colour (W). As one traverses the hierarchy from the subordinate, to basic, to superordinate levels of abstraction, the cardinality of the concept-defining sets C_i drops from four, to two-three, to one, respectively. Hence, each parent concept in such a hierarchy is an abstraction (a subset) over its children and over the set of original visual primitives. We provide a more formal definition of this intuitive notion in Sec.2.1 .

SCAN is able to learn such an implicit hierarchy of concepts through a multi-stage process. First, we use the previously published β-VAE (Higgins et al., 2017a) to learn a set of independent representational primitives through unsupervised exposure to the visual data. This is equivalent to learning a disentangled (factorised and interpretable) representation of the independent ground truth "generative factors" of the data (Bengio et al., 2013). Next the model has to discover and learn meaningful abstractions over these disentangled primitives, since this is how we define concepts. This is done through a sample-efficient symbol association process. In particular, the model is exposed to a small number of symbol-image pairs that apply to a particular concept (e.g. a few example images of an apple paired with the symbol "apple"). SCAN learns the meaning of the concept "apple" by identifying the set of visual primitives that all example apples have in common (e.g. all observed apples are small, round and red). The corresponding symbol ("apple") then becomes a "pointer" to the newly acquired concept {small, round, red} - a way to access and manipulate the concept without having to know its exact representational form. Our approach does not make any assumptions about how these symbols are encoded, which allows SCAN to learn multiple referents to the same concept (synonyms).

Once a concept is acquired, it should be possible to use it for bi-directional inference: the model should be able to generate diverse visual samples that correspond to a particular concept (*sym2img*) and vice versa (*img2sym*). Since the projection from the space of visual primitives to the space of concepts (red arrow in Fig. 2) involves abstraction and hence a potential loss of information, it is important to fill that information back in when moving backwards from the space of concepts to that of visual primitives (blue arrow in Fig. 2) when doing sym2img inference. In our setup, concepts are defined in terms of a set of relevant visual primitives (e.g. colour, shape and size for "apple"). This

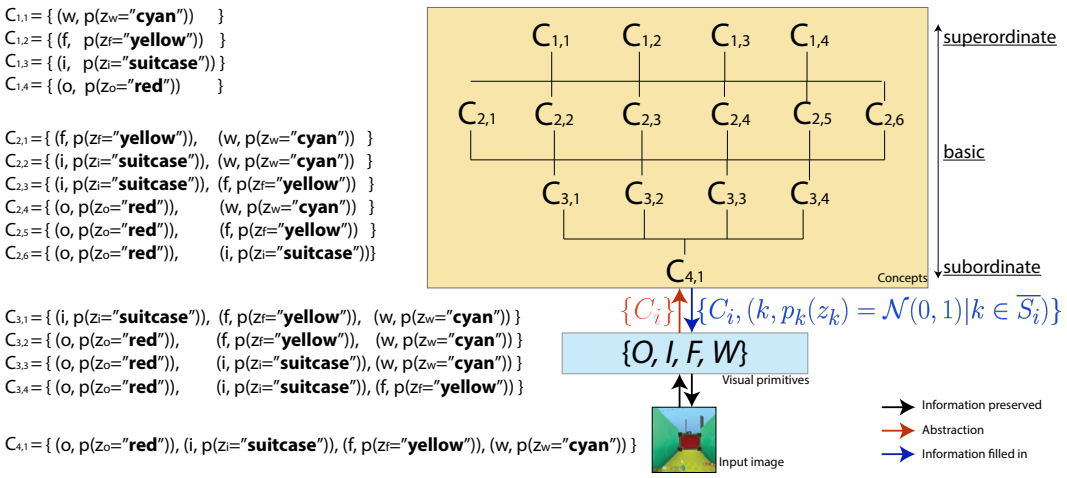

Figure 2: Schematic of an implicit concept hierarchy reflecting the underlying structure in the DeepMind Lab dataset (Beattie et al., 2016). The hierarchy builds upon four visual primitives: object identity (I), object colour (O), floor colour (F) and wall colour (W), which we represent as random variables. Any provided input image $\mathbf{x}$, can thus be represented as a multimodal probability distribution $q(\mathbf{z}|\mathbf{x})$. Alternatively it can be represented as a set of tuples of visual primitives $\{(o, q(z_o|\mathbf{x})), (i, q(z_i|\mathbf{x})), (f, q(z_f|\mathbf{x})), (w, q(z_w|\mathbf{x}))\}$. Concepts are abstractions over this representation that form an implicit hierarchy. In SCAN, an abstraction is represented as a set of relevant primitives, with a narrowly peaked distribution around the correct value (shown in bold), while the remaining irrelevant visual primitives learn to default to the broad prior distribution. Each image can be described using concepts from different levels of the hierarchy. Each parent concept is an abstraction over its children (specified using a subset of the child's primitives). This is related to the idea of different levels (superordinate/basic/subordinate) of cognitive categorisation in psychology (Rosch, 1978). Given certain nodes in the concept tree, one can traverse the hierarchy to other unknown nodes through concept manipulation operations, e.g. {red, suitcase} = {red, suitcase, yellow_floor} IGNORE {yellow_floor}

leaves a set of irrelevant visual attributes (e.g. lighting, position, background) to be "filled in". We do so by defaulting them to their respective priors, which ensures high diversity of image samples corresponding to each concept when asked to imagine instances of that concept. The same process applies to img2sym inferences.

In the final stage of training we teach SCAN logical recombination operators: AND (corresponding to a set union of relevant primitives), IN COMMON (corresponding to set intersection) and IGNORE (corresponding to set difference), by pairing a small number of valid visual examples of recombined concepts with the respective operator names. Once the meaning of the operators has been successfully learned, SCAN can exploit the compositionality of the acquired concepts, and traverse the (implicit) underlying concept hierarchy by manipulating and recombining existing concepts in novel ways.

As far as we are aware, no framework currently exists that is directly equivalent to SCAN. Past relevant literature can broadly be split into three categories: 1) Bayesian models that try to mimic fast human concept learning (Tennenbaum, 1999; Lake et al., 2015); 2) conditional generative models that aim to generate faithful images conditioned on a list of attributes or other labels (Reed et al., 2016; Kingma et al., 2014; Yan et al., 2016; Sohn et al., 2015; Pandey & Dukkipati, 2017) ; and 3) multimodal generative models that aim to embed visual and symbolic inputs in a joint latent space in order to be able to run bi-directional inferences (Srivastava & Salakhutdinov, 2014; Suzuki et al., 2017; Pu et al., 2016; Wang et al., 2016). While all of these approaches have their strengths, none of them are capable of sample efficient compositional concept learning as defined above. Bayesian models by (Tennenbaum, 1999; Lake et al., 2016) can learn from few examples, but are not fully grounded in visual data. Conditional and joint multimodal models are fully grounded in visual data, however they require a large number of image-symbol pairs for training (apart from the model by Srivastava & Salakhutdinov (2014), which has its own problem of prohibitively slow MCMC

sampling). More importantly though, unlike SCAN, all of the approaches mentioned learn flat unstructured representations that lack hierarchy-spanning compositionality.

One approach that comes closer than others to ours is that of Vedantam et al. (2017), released concurrently with our work. While the problem definition for compositional concept learning in Vedantam et al. (2017) is similar to ours, it lacks the data efficiency and strong compositionality characteristic of our SCAN proposal. In particular, their model is a variation on the joint multimodal generative model approaches which, in our opinion, have inherent difficulties with respect to concept learning. We believe that visual (x) and language (y) domains are inherently asymmetric, where language corresponds to information-removing abstractions, rather than to full representations of images. Thus, they are necessarily asymmetric in terms of information content I, where $I(x) \geqslant I(y)$. Intuitively, an image that matches a particular description contains extra information that is not specified by the text (e.g. an image that matches the description "green round object" contains other information, such as the scale, the position or the lighting of the object, as well as all the other idiosyncrasies of the specific image instance). Conversely, even the most detailed description of the image (given that the description is restricted to refer purely to the visual content) can never contain more information than the image itself. Hence, we argue that any approach that tries to learn a joint embedding for both domains will have to make compromises between the different encoding requirements of the image and symbol domains. This is likely to affect the ability of such approaches to learn a representation that has the right compositional hierarchical structure necessary for concept encoding.

To summarise, our proposed SCAN is capable of learning a compositional representation of visual concepts in a fully grounded manner and with very little supervision, whereby the conceptual meaning of symbols is discovered from a small subset of image-symbol pairs (as few as five visual examples per symbol). We highlight the data efficiency of our approach because unlabelled visual data tends to be cheap to acquire, while coherent symbolic referents are harder to source. It also brings our approach closer to human word learning than existing alternatives, as humans acquire the meaning of words through a combination of an incessant stream of unsupervised visual data, and occasional pairing with word labels. After training, SCAN can perform multimodal (visual and symbolic) bi-directional inference and generation (sym2img and img2sym) with high accuracy and diversity, outperforming all baselines. Unlike any other approaches, SCAN can also make use of the learnt logical recombination operations to imagine and learn novel concepts, thereby reaching new nodes within the implicit hierarchy of concepts that have never been experienced during training. Crucially, this allows SCAN to break out of its limited training data distribution to imagine and learn new concepts from purely symbolic instructions. Due to the sample efficiency and the limited number of assumptions in our approach, the compositional abstract hierarchical representations learnt by SCAN should be immediately applicable within a large set of broader problem domains, including reinforcement learning, classification, control and planning.

2 Framework

In this section we first formalise the notion of concepts and logical recombination operators over them, before introducing SCAN: a way to implement these ideas within a neural network architecture capable of learning grounded visual concepts from raw pixels and a small number of image-symbol pairs.

2.1 Formalising concepts

Intuitively we have proposed that concepts are abstractions over visual representational primitives. Hence, in order to formally define concepts we first define the visual representations used to ground the concepts as tuples of random variables of the form $(Z_1, ..., Z_K)$, where $\{1, ..., K\}$ is the set of indices of the independent latent factors sufficient to generate the visual input $\mathbf{x}$, and Z_k is a random variable. The set $\mathbb{R}^K$ of all such tuples is a K-dimensional visual representation space.

We define a concept C_i in such a K-dimensional representation space as a set of assignments of probability distributions to the random variables Z_k, with the following form:

$$C_i = \{(k, p_k^i(z_k)) \mid k \in S_i\}$$

where $S_i \subseteq \{1, ..., K\}$ is the set of visual latent primitives that are relevant to concept C_i and $p_k^i(z_k)$ is a probability distribution specified for the particular visual primitive Z_k.

Since S_i are subsets of $\{1, ..., K\}$, concepts are abstractions over the K-dimensional visual representation space. To generate a visual sample corresponding to a concept C_i, it is necessary to fill in the details that got abstracted away during concept learning. This corresponds to the probability distributions $\{p_k(z_k)|k \in \overline{S_i}\}$, where $\overline{S_i} = \{1, ..., K\} \setminus S_i$ is the set of visual latent primitives that are irrelevant to the concept C_i. In SCAN we set these to the unit Gaussian prior: $p_k(z_k) = \mathcal{N}(0, 1),\ \forall k \in \overline{S_i}$.

If C_1 and C_2 are concepts, and $C_1 \subset C_2$, we say that C_1 is *superordinate* to C_2, and C_2 is *subordinate* to C_1. Two concepts C_1 and C_2 are *orthogonal* if $S_1 \cap S_2 = \varnothing$. The *conjunction* of two orthogonal concepts C_1 and C_2 is the concept $C_1 \cup C_2$ ("C_1 AND C_2"). The *overlap* of two non-orthogonal concepts C_1 and C_2 is the concept $C_1 \cap C_2$ ("C_1 IN COMMON C_2"). The *difference* between two concepts C_1 and C_2, where $C_1 \subset C_2$ is the concept $C_2 \setminus C_1$ ("C_2 IGNORE C_1").

2.2 Learning visual representational primitives

The discovery of the generative structure of the visual world is the goal of disentangled factor learning literature (Bengio et al., 2013). In this work we build SCAN on top of β-VAE, a state of the art approach to unsupervised visual disentangled factor learning. β-VAE is a modification of the variational autoencoder (VAE) framework (Kingma & Welling, 2014; Rezende et al., 2014) that introduces an adjustable hyperparameter β to the original VAE objective:

$$\mathcal{L}_x(\theta_x, \phi_x; \mathbf{x}, \mathbf{z}_x, \beta_x) = \mathbb{E}_{q_{\phi_x}(\mathbf{z}_x|\mathbf{x})}[\log p_{\theta_x}(\mathbf{x}|\mathbf{z}_x)] - \beta_x\ D_{KL}(q_{\phi_x}(\mathbf{z}_x|\mathbf{x})\ ||\ p(\mathbf{z}_x)) \tag{1}$$

where ϕ, θ parametrise the distributions of the encoder and the decoder respectively. Well chosen values of β (usually $\beta > 1$) result in more disentangled latent representations $\mathbf{z}$ by setting the right balance between reconstruction accuracy, latent channel capacity and independence constraints to encourage disentangling. For some datasets, however, this balance is tipped too far away from reconstruction accuracy. In these scenarios, disentangled latent representations $\mathbf{z}$ may be learnt at the cost of losing crucial information about the scene, particularly if that information takes up a small proportion of the observations $\mathbf{x}$ in pixel space. We encountered this issue in the DeepMind Lab (Beattie et al., 2016) environment. The same problem was encountered and solved in Higgins et al. (2017b). We adopt their solution, which suggests replacing the pixel level log-likelihood term $\mathbb{E}_{q_{\phi_x}(\mathbf{z}|\mathbf{x})}[\log p_{\theta_x}(\mathbf{x}|\mathbf{z})]$ in Eq. 1 with an L2 loss in the high-level feature space of a denoising autoencoder (DAE) (Vincent et al., 2010) trained on the same data (see Fig. 1C for model architecture). The resulting β-VAE$_{DAE}$ architecture optimises the following objective function:

$$\mathcal{L}_x(\theta_x, \phi_x; \mathbf{x}, \mathbf{z}_x, \beta_x) = \mathbb{E}_{q_{\phi_x}(\mathbf{z}_x|\mathbf{x})}\ \|J(\hat{\mathbf{x}}) - J(\mathbf{x})\|_2^2 - \beta_x\ D_{KL}\big(q_{\phi_x}(\mathbf{z}_x|\mathbf{x})\ ||\ p(\mathbf{z}_x)\big) \tag{2}$$

where $\hat{\mathbf{x}} \sim p_{\theta_x}(\mathbf{x}|\mathbf{z})$ and $J : \mathbb{R}^{W \times H \times C} \rightarrow \mathbb{R}^N$ is the function that maps images from pixel space with dimensionality $W \times H \times C$ to a high-level feature space with dimensionality N given by a stack of DAE layers up to a certain layer depth. Note that this adjustment means that we are no longer optimising the variational lower bound, and β-VAE$_{DAE}$ with $\beta = 1$ loses its equivalence to the original VAE framework. In this setting, we interpret β as a mixing coefficient that balances the capacity of the latent channel $\mathbf{z}$ of β-VAE$_{DAE}$ against the pressure to match the high-level features within the DAE.

2.3 Learning visual concepts

As mentioned in Sec. 2.1, we define concepts as abstractions over visual representational primitives. In the previous section we have described how such visual primitives may be acquired by β-VAE[1]. This

[1]For the rest of the paper we use the term β-VAE to refer to β-VAE$_{DAE}$.

section describes how our proposed SCAN framework (Fig. 1A) exploits the particular parametrisation of the visual building blocks to learn an implicit hierarchy of visual concepts as formalised in Sec. 2.1. Similarly to β-VAE, SCAN is based on the VAE framework, but with certain modifications motivated hereafter.

As mentioned in Sec. 2.1, we want each symbol to refer to a concept grounded in a set of relevant visual primitives, while the irrelevant ones are set to their prior distribution. Relevant primitives are those that the symbol actually refers to (e.g. "apple" refers to objects of a particular shape, colour and size), while irrelevant primitives can be thought of as everything else (e.g. background, lighting, position). We suggest parametrising each concept as a multivariate Gaussian distribution with a diagonal covariance matrix, where variables corresponding to the relevant factors should have narrow distributions, while those corresponding to the irrelevant factors should default to the wider prior unit Gaussian distribution.

One way to implement this is to initialise the space of abstract concepts (the latent space $\mathbf{z}_y$ of SCAN) to be structurally identical to the space of visual primitives (the latent space $\mathbf{z}_x$ of β-VAE). Both spaces are parametrised as multivariate Gaussian distributions with diagonal covariance matrices, and $|\mathbf{z}_y| = |\mathbf{z}_x| = K$. To ground abstract concepts $\mathbf{z}_y$ in the visual primitives $\mathbf{z}_x$, we aim to minimise the KL divergence between the two distributions. In addition, to ensure that those SCAN latents z_y^k that correspond to the irrelevant building blocks z_x^k in the β-VAE learn to default to the unit Gaussian prior $p(\mathbf{z}_x)$, it is important to minimise the forward KL divergence $D_{KL}(\mathbf{z}_x||\mathbf{z}_y)$, rather than the mode picking reverse KL divergence $D_{KL}(\mathbf{z}_y||\mathbf{z}_x)$ (see Fig. 1B for a schematic of the differences). To combine both requirements, SCAN is trained by minimising

$$\begin{aligned}\mathcal{L}_y(\theta_y, \phi_y; \mathbf{y}, \mathbf{z}_y, \beta_y, \lambda) =& \mathbb{E}_{q_{\phi_y}(\mathbf{z}_y|\mathbf{y})}[\log p_{\theta_y}(\mathbf{y}|\mathbf{z}_y)] - \beta_y \, D_{KL}(q_{\phi_y}(\mathbf{z}_y|\mathbf{y}) \,||\, p(\mathbf{z}_y)) \\ & - \lambda \, D_{KL}(q_{\phi_x}(\mathbf{z}_x|\mathbf{x}) \,||\, q_{\phi_y}(\mathbf{z}_y|\mathbf{y}))\end{aligned} \tag{3}$$

where $\mathbf{y}$ is symbol inputs, $\mathbf{z}_y$ is the latent space that learns to represent concepts, $\mathbf{z}_x$ is the latent space of the pre-trained β-VAE containing the visual primitives which ground the abstract concepts $\mathbf{z}_y$, and $\mathbf{x}$ are example images that correspond to the concepts $\mathbf{z}_y$ activated by symbols $\mathbf{y}$. A weighted interaction of the two KL terms in Eq. 3 can be tuned to ensure that $q(\mathbf{z}_y)$ does not deviate too much from the prior $p(\mathbf{z}_y)$, while learning to be an abstraction over $q(\mathbf{z}_x)$. In practice we find that $\beta_y = 1, \lambda = 10$ works well.

The SCAN architecture does not require any assumptions on the encoding of the symbol inputs $\mathbf{y}$. In this paper we use a k-hot encoding for $\mathbf{y}$, where $k \leq K$ is the number of factors *specified* by a symbol (relevant to a concept) out of K total data generative factors. Other possible encoding schemes for $\mathbf{y}$ can also be used, including word embedding vectors (Mikolov et al., 2013) (the encoder and decoder networks of SCAN can be implemented as MLPs or RNNs), or even entirely random vectors.

Once trained, SCAN allows for bi-directional inference and generation (img2sym and sym2img). In order to generate visual samples that correspond to a particular concept (sym2img), we infer the concept $\mathbf{z}_y$ by presenting an appropriate symbol $\mathbf{y}$ to the inference network of SCAN. One can then sample from the inferred concept $q_{\phi_y}(\mathbf{z}_y|\mathbf{y})$ and use the generative part of β-VAE to visualise the corresponding image samples $p_{\theta_x}(\mathbf{x}|\mathbf{z}_y)$. SCAN can also be used to infer a description of an image in terms of the different learnt concepts via their respective symbols. To do so, an image $\mathbf{x}$ is presented to the inference network of the β-VAE to obtain its description in terms of the visual primitives $\mathbf{z}_x$. One then uses the generative part of the SCAN to sample descriptions $p_{\theta_y}(\mathbf{y}|\mathbf{z}_x)$ in terms of symbols that correspond to the previously inferred visual building blocks $q_{\phi_x}(\mathbf{z}_x|\mathbf{x})$.

2.4 Learning of concept recombination operators

Sec. 2.3 described how SCAN can learn grounded visual concepts from across the various arbitrary levels of their respective implicit hierarchy (as shown in Fig. 2). This section describes how such concepts can be manipulated to traverse the full hierarchy. To do so, we use symbolic instructions in combination with logical concept manipulation operators. The operators are AND, IN COMMON and IGNORE as formally defined in Sec. 2.1. These operators are learnt by presenting the model with a small number of images paired with a matching recombination instruction (e.g. an image of an blue object paired with "blue suitcase IGNORE suitcase"). Once learnt, the operators can

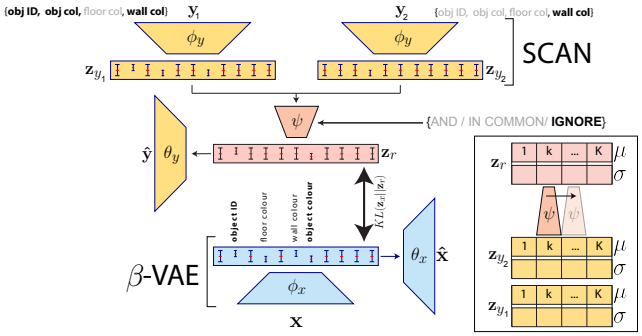

Figure 3: Learning AND, IN COMMON or IGNORE recombination operators with a SCAN model architecture. Inset demonstrates the convolutional recombinator operator that takes in $\{\mu^k_{y_1}, \sigma^k_{y_1}; \mu^k_{y_2}, \sigma^k_{y_2}\}$ and outputs $\{\mu^k_r, \sigma^k_r\}$.

be used to traverse previously unexplored parts of the concepts hierarchy. For example, a new node corresponding to the concept "blue small" can be traversed through the following instructions: "blue AND small" (going down the hierarchy from more general to more specific), "blueberry IN COMMON bluebell" (going up the hierarchy from more specific to more general) or "blueberry IGNORE round" (also going up the hierarchy).

We parameterise the operators as a conditional convolutional module (Fig. 3) that accepts two multivariate Gaussian distributions $\mathbf{z}_{y_1} = q_{\phi_y}(\cdot|\mathbf{y}_1)$ and $\mathbf{z}_{y_2} = q_{\phi_y}(\cdot|\mathbf{y}_2)$, and a one-hot conditioning vector $\mathbf{h}$, which is the symbolic label for the corresponding operator. The Gaussian distributions are inferred from two corresponding input symbols $\mathbf{y}_1$ and $\mathbf{y}_2$, respectively, using a pre-trained SCAN. The convolutional module strides over the parameters of each matching component $z^k_{y_1}$ and $z^k_{y_2}$ one at a time and outputs the corresponding parametrised component z^k_r of a recombined multivariate Gaussian distribution $\mathbf{z}_r$ with a diagonal covariance matrix.[2] The resulting $\mathbf{z}_r$ lives in the same space as $\mathbf{z}_y$ and hence corresponds to a node of the implicit hierarchy of visual concepts. Thus, it is also grounded via the visual building blocks $\mathbf{z}_x$. Hence, all the properties of concepts $\mathbf{z}_y$ discussed in the previous section also hold for $\mathbf{z}_r$. The conditioning on the particular recombination operator $\mathbf{h}$ is implemented by a tensor product operation, and the full recombination module is trained by minimising

$$\mathcal{L}_r(\psi; \mathbf{z}_x, \mathbf{z}_r) = D_{KL}\left[q_{\phi_x}(\mathbf{z}_x|\mathbf{x}_i) \,\middle|\middle|\, q_\psi\left(\mathbf{z}_r \mid q_{\phi_y}(\mathbf{z}_{y_1}|\mathbf{y}_1), q_{\phi_y}(\mathbf{z}_{y_2}|\mathbf{y}_2), \mathbf{h}\right)\right] \tag{4}$$

where $q_{\phi_x}(\mathbf{z}_x|\mathbf{x}_i)$ is the inferred latent distribution of the β-VAE given a seed image $\mathbf{x}_i$ that matches the specified symbolic description.

2.5 Fast learning of new concepts from symbolic instructions

By teaching logical operators to SCAN as described in Sec. 2.4, we give it the ability to construct and imagine new concepts that it may have never seen during training. It does not, however, allow the model to extend its vocabulary, which would require attaching new symbols to these newly constructed concepts. In this section we describe how we can teach SCAN to associate a new symbol to a novel grounded visual concept using purely symbolic instructions (without providing any visual examples).

We aim to continue teaching SCAN new concepts, including those defined purely through symbolic instructions, without disrupting the model's understanding of existing symbols and logical operators (avoiding catastrophic interference). To to so, we instantiate two copies of the pre-trained SCAN, SCAN_y and SCAN_{IS}, which intermittently sync their parameters ($\phi_{\text{IS}} \rightarrow \phi_y$ and $\theta_{\text{IS}} \rightarrow \theta_y$) (see Fig. 4). During training new concepts are introduced to SCAN_{IS} in the form of symbolic instructions given to SCAN_y, such as {"pom" IS "blueberry" IN COMMON "bluebell"}. In this example "pom"

[2]We also tried a closed form implementation of recombination operators (weighted sum or mean of the corresponding Gaussian components $z^k_{y_1}$ and $z^k_{y_2}$). We found that the learnt recombination operators were more precise.

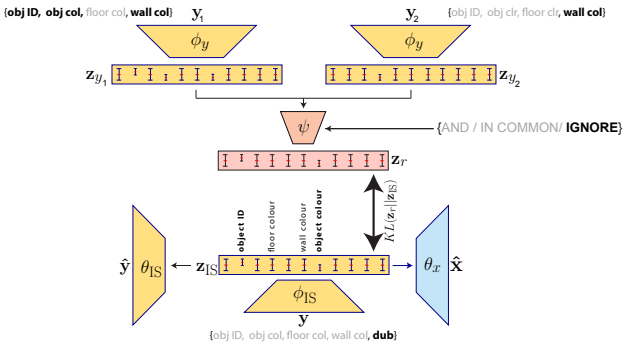

Figure 4: Learning IS operator with a SCAN model architecture.

is a new made-up lexicographic symbol that we want SCAN_{IS} to learn to associate to the new concept "blue small". This is done by minimising the following objective function:

$$\begin{aligned}\mathcal{L}_{\text{IS}}(\theta_{\text{IS}}, \phi_{\text{IS}}; \mathbf{z}_{\text{IS}}, \mathbf{z}_r, \beta_{\text{IS}}, \lambda_{\text{IS}}) = &\mathbb{E}_{q_{\phi_{\text{IS}}}(\mathbf{z}_{\text{IS}}|\mathbf{y})}[\log p_{\theta_{\text{IS}}}(\mathbf{y}|\mathbf{z}_{\text{IS}})] - \beta_{\text{IS}}\, D_{KL}\left(q_{\phi_{\text{IS}}}(\mathbf{z}_{\text{IS}}|\mathbf{y}) \,||\, p(\mathbf{z}_y)\right) \\ &- \lambda_{\text{IS}}\, D_{KL}(q_\psi(\mathbf{z}_r) \,||\, q_{\phi_{\text{IS}}}(\mathbf{z}_{\text{IS}}|\mathbf{y}))\end{aligned} \tag{5}$$

where $q_\psi = q_\psi(\cdot|q_{\phi_y}(\mathbf{z}_{y_1}|\mathbf{y}_1), q_{\phi_y}(\mathbf{z}_{y_2}|\mathbf{y}_2), \mathbf{h})$ is the output of the pre-trained recombination module described in Sec. 2.4 given the latent states $\mathbf{z}_{y_1}$ and $\mathbf{z}_{y_2}$ inferred by the SCAN_y encoder ϕ_y, as well as the conditioning recombination operator $\mathbf{h}$. See Sec. A.2 in Supplementary Materials for more training details.

3 Experiments

We evaluate the performance of SCAN and its extensions (symbol-instructed imagination and concept learning) on a DeepMind Lab (Beattie et al., 2016) dataset (see Fig. 11 in Supplementary Materials for example frames). This dataset was also used to compare the performance of SCAN to that of baselines - a SCAN like architecture trained using an unstructured visual encoder, and one of the latest multimodal joint density models, the JMVAE (Suzuki et al., 2017). DeepMind Lab frames were collected from a static viewpoint situated in a room containing a single object. The room was generated by specifying four factors of variation: wall colour, floor colour, object colour with 16 possible values each, and object identity with 3 possible values: hat, ice lolly and suitcase. Other factors of variation were also added to the dataset by the DeepMind Lab engine, such as the spawn animation, horizontal camera rotation and the rotation of objects around the vertical axis. We split the dataset into two subsets. One was used for training the models, while the other one contained a held out set of 301 unique concepts used to evaluate the models' ability to imagine and learn new concepts.

In this section we answer the following questions:

1. Can SCAN learn visual concepts across the implicit hierarchy from a small number of symbol-image pairs?
2. How does the nature of the concepts learnt by SCAN change throughout training as the model is exposed to more and more examples that apply to a particular symbol?
3. Can SCAN learn multiple symbolic referents to the same concept (synonyms)?
4. Can SCAN perform bi-directional inference with sufficient variability? And how does its performance compare to baselines?
5. Can SCAN learn recombination operators from few samples?
6. Can these recombination operators be used to imagine novel concepts?
7. Can SCAN be taught new concepts from purely symbolic instructions?

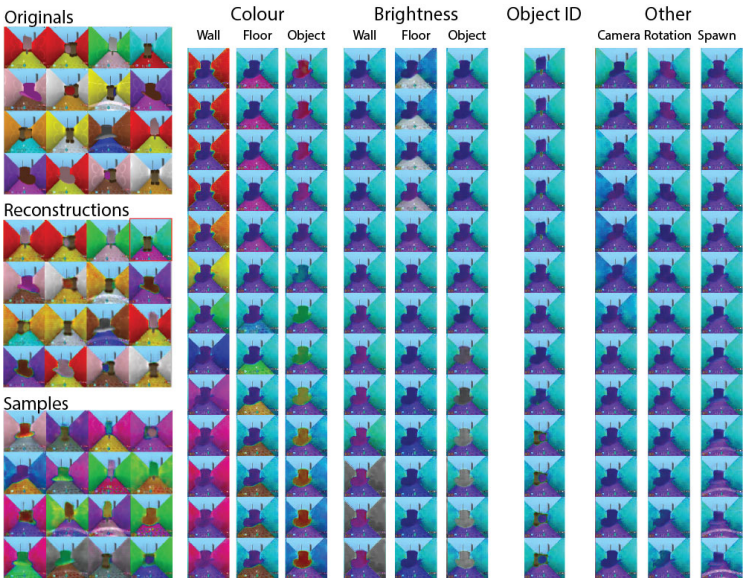

Figure 5: Reconstructions, samples and latent traversals of β-VAE trained to disentangle the data generative factors of variation within the DeepMind Lab dataset. For the latent traversal plots we sampled the posterior, then visualised β-VAE reconstructions while resampling each latent unit one at a time in the $[-3, 3]$ range while keeping all other latents fixed to their originally sampled values. This process helps visualise which data generative factor each latent unit has learnt to represent. Note that due to the nature of reconstruction cost of β-VAE, which is taken in the high level feature space of a pre-trained DAE, the model sometimes confuses colour (e.g. red floor is reconstructed as magenta, see the top right reconstruction).

Visual representation learning Our proposed framework that motivates the architectural choices in SCAN relies on the presence of structured visual primitives. Hence, we first investigate whether β-VAE trained in an unsupervised manner on the visually complex DeepMind Lab dataset has discovered a disentangled representation of all its data generative factors. As can be seen in Fig. 5, SCAN has learnt to represent each of the object-, wall-, and floor-colours, using two latents – one for hue and one for brightness. Learning a disentangled representation of colour is challenging, but we were able to achieve it by projecting the input images from RGB to HSV space, which is better aligned with human intuitions of colour (see Sec. A.2 in Supplementary Materials for discussion). We noticed that β-VAE confused certain colours (e.g. red floors are reconstructed as magenta, see the top right in Fig. 5). We speculate that this is due to taking the reconstruction cost in the high level feature space of the DAE (see Fig. 5), as opposed to pixel space.

Evaluation of concept understanding In this section we demonstrate that SCAN is capable of learning the meaning of new concepts from very few image-symbol pairs. We evaluate the model's concept understanding through qualitative analysis of sym2img and img2sym samples. SCAN was taught using a random subset of 100 out of 12,288 possible concepts, by sampling from the full implicit hierarchy defined by the DeepMind Lab dataset (these concepts specify between one and four factors of variation, e.g. "white object", or "white hat in a room with blue walls and a yellow floor". See Fig. 2 for a schematic of the hierarchy). We also included some synonyms in the training set by allowing a number of different symbols to refer to the same concept (e.g. a blue wall may be described by symbols "blue_wall", "bright_blue_wall" or "blue_wall_synonym"). SCAN saw twenty example images per concept.

Fig. 6A shows samples drawn from SCAN when asked to imagine a bigram concept "white suitcase", a trigram concept "white suitcase, blue_wall", or a four-gram "white suitcase, blue_wall, magenta_floor". Note that the latter is a concept drawn from the held-out test set that neither β-VAE nor SCAN have ever seen during training, and the first two concepts are novel to SCAN, but have been experienced by β-VAE. It is evident that the model demonstrates a good understanding of all three concepts, producing visual samples that match the meaning of the concept, and showing good variability over the irrelevant factors. Confusions do sometimes arise due to the stochastic sampling

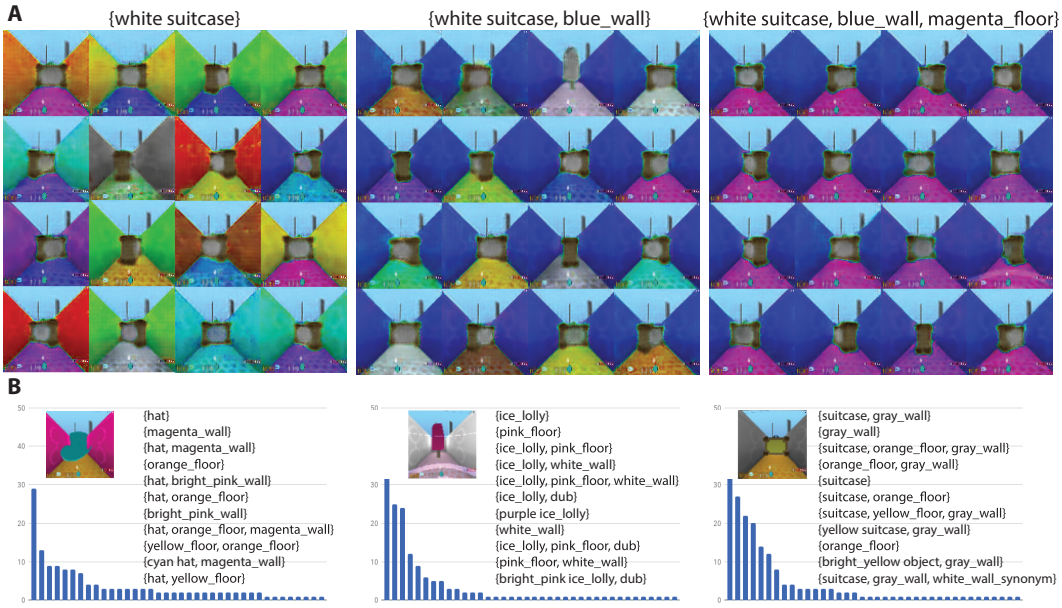

Figure 6: **A**: sym2img inferences with "white suitcase", "white suitcase, blue wall", and "white suitcase, blue wall, magenta floor" as input. The latter one is a concept that the model has never seen during training, neither visually not symbolically. All samples consistently match the meaning of the concepts, while showing good variability in all irrelevant visual attributes. **B**: when presented with an image, SCAN is able to describe it in terms of all concepts it has learnt, including synonyms (e.g. "dub", which corresponds to "ice lolly, white wall"). The histograms show the distributions of unique concepts the model used to describe each image, most probable of which are printed in descending order next to the corresponding image. The few confusions SCAN makes are intuitive to humans too (e.g. confusing orange and yellow colours).

process (e.g. one of the suitcase samples is actually an ice lolly). However, these confusions are mostly similar to ones that humans would make, too (e.g. sampling dark and light blue hats given an instruction to imagine a "blue hat"). Fig. 6B demonstrates that the same model can also correctly describe an image. The labels are mostly consistent with the image and display good diversity (SCAN is able to describe the same image using different symbols including synonyms). The few confusions that SCAN does make are between concepts that are easily confusable for human too (e.g. red, orange and yellow colours).

Evolution of concept learning In this section we take a closer look at the evolution of concept grounding during SCAN training. In Sec. 2.1 we suggested that concepts should be grounded in terms of relevant factors, while the irrelevant visual primitives should be set to the unit Gaussian prior. For example, the set of relevant primitives for the concept "apple" might include shape, size and colour, while the set of irrelevant primitives might include background or lighting. Within SCAN each concept is implemented as a multivariate diagonal Gaussian distribution $\mathbf{z}_y$ that matches the latent space of visual primitives $\mathbf{z}_x$. A subset of latent units z_y^k that correspond to the relevant visual primitives z_x^k have to learn low inferred standard deviations σ_y^k (highly specified), while the rest of the latents z_y^k have to default to their unit Gaussian prior $p(\mathbf{z}_y)$ (unspecified). We visualise this process by teaching SCAN the meaning of the concept "cyan wall" within the DeepMind Lab dataset using a curriculum of fifteen progressively more diverse visual examples (see bottom row of Fig. 7). After each five examples we test the model's understanding of the concept by drawing visual samples corresponding to "cyan wall" (top row of Fig. 7). We also plot the average inferred specificity of all 32 latent units z_y^k during training (Fig. 7, right). It can be seen that the number of specified latents z_y^k drops from six, to four, to two as the diversity of visual examples seen by SCAN increases. The remaining two highly specified latents z_y^k correctly correspond to the visual primitives $\mathbf{z}_x$ representing

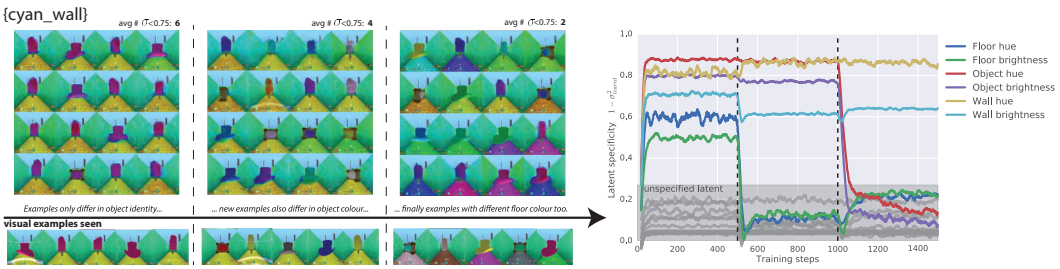

Figure 7: Evolution of understanding of the meaning of concept "cyan wall" as SCAN is exposed to progressively more diverse visual examples. **Left**: top row contains three sets of visual samples (sym2img) generated by SCAN after seeing each set of five visual examples presented in the bottom row. **Right**: average inferred specificity of concept latents z_y^k during training. Vertical dashed lines correspond to the vertical dashed lines in the left plot and indicate a switch to the next set of five more diverse visual examples. 6/32 latents z_y^k and labelled according to their corresponding visual primitives in $\mathbf{z}_x$.

wall hue and brightness. The change in the number of specified units is inversely correlated with the diversity of visual samples.

Quantitative comparison to baselines This section provides quantitative comparison of the accuracy and diversity of sym2img inferences of SCAN to those of baseline models - a SCAN coupled with an unstructured visual representation (SCAN_U), and a recent example of a multimodal generative model JMVAE (Suzuki et al., 2017). It is important to evaluate models in terms of both metrics, since they measure different aspects of the models' performance. Since concepts correspond to both relevant and irrelevant visual primitives (see Sec. 2.1), full concept understanding is only achieved if the models produce samples that have both high accuracy and high diversity. Visual accuracy measures whether the model samples include the visual attributes relevant to the concepts (e.g. samples of a "blue suitcase" should contain blue suitcases). Diversity measure quantifies the variety of samples in terms of the visual attributes irrelevant to the concepts (e.g. samples of blue suitcases should include a high diversity of wall colours and floor colours).

	ACCURACY		DIVERSITY
MODEL	SEEN	NOVEL	SEEN
JMVAE	0.79	0.7	5.3
SCAN_U	0.27	0.26	3.77
SCAN	**0.83**	**0.85**	**2.72**

Table 1: Quantitative results comparing the accuracy and diversity of visual samples produced through sym2img inference by SCAN and two baselines: a SCAN with unstructured vision (SCAN_U) and a recent joint multimodal embedding model, JMVAE (Suzuki et al., 2017). Higher values of the accuracy metric and lower values of the diversity metric are better.

For visual accuracy and diversity evaluation we use a pre-trained classifier achieving 99% average accuracy over all data generative factors in DeepMind Lab. While its performance drops when evaluating model samples due to the inevitable domain shift, it serves as a good measure of relative accuracy of SCAN and baseline samples. Since some colours in the dataset are hard to differentiate even to humans (e.g. yellow and orange), we use top-3 accuracy for colour related factors. We evaluate the diversity of visual samples by estimating the KL divergence of the inferred factor distribution with the flat prior: $KL(u(\mathbf{y}_i)||p(\mathbf{y}_i))$, where $p(\mathbf{y}_i)$ is the joint distribution over the factors irrelevant to the ith concept $i \in \overline{S_i}$ (inferred by the classifier) and $u(\mathbf{y}_i)$ is the equivalent flat distribution (i.e., with each factor value having equal probability). See Sec. A.1 in Supplementary Materials for more details.

The accuracy metric was calculated using two test sets: the *seen* set consisting of 16,000 samples corresponding to the concepts within the train split of the dataset (remember that all models saw less

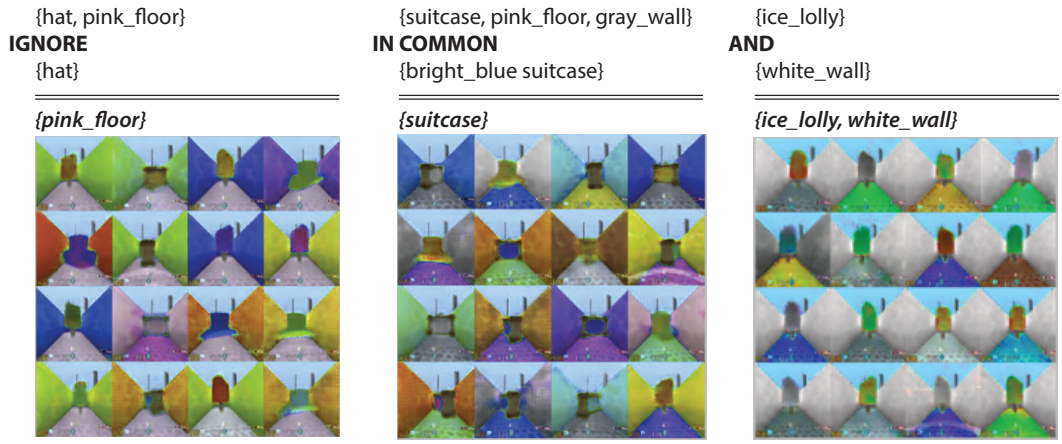

Figure 8: Visual samples produced by SCAN when instructed with a novel concept recombination. The samples consistently match the expected ground truth recombined concept, while maintaining high variability in the irrelevant visual primitives. Recombination instructions are used to imagine concepts that have never been seen during SCAN training. **Left**: samples for IGNORE; **Middle**: samples for IN COMMON; **Right**: samples for AND.

than 1% of them during training); and the *novel* set consisting of 4,816 samples corresponding to the set of 301 held out concepts that the models never experienced either visually or symbolically. The diversity metric was only calculated using the *seen* set, since all concepts in the *novel* set were fully specified and hence the set of irrelevant factors $\overline{S_i}$ was empty.

Tbl. 1 demonstrates that SCAN coupled with disentangled vision outperforms both baselines in terms of accuracy and diversity of samples. SCAN_U struggles on both metrics, because by definition it learns an entangled representation of the ground truth factors of variation where the posterior $q(\mathbf{z}_y|\mathbf{y})$ is not constrained to lie within the unit ball of the prior $p(\mathbf{z}_y)$. Since each data generative factor is encoded by an overlapping set of latents z_y^k taking on a wide range of values ($\mu_y^k \in [-5, 5]$, $\sigma \approx 0.1$), the additional KL term of SCAN ($KL(\mathbf{z}_x||\mathbf{z}_y)$ in Eq. 3) is unable to learn well defined sets of relevant and irrelevant visual primitives. Furthermore, the KL term applied to the unstructured visual representation of SCAN_U results in equally wide learnt distributions for all z_y^k, which means that $\mathbf{z}_y$ often falls outside of the generative distribution $q_\theta(\mathbf{x}|\mathbf{z}_y)$ learnt by SCAN_U, resulting in bad visual samples with low accuracy (see Sec. A.4 for more details). Note that the bad quality of SCAN_U samples inadvertently results in an improvement in our diversity measure. Such samples are hard to classify, which produces a relatively flat classifier distribution $p(\mathbf{y}_i)$ close to the uniform prior $u(\mathbf{y}_i)$, and hence an improvement in $KL(u(\mathbf{y}_i)||p(\mathbf{y}_i))$ which acts as our measure of diversity.

JMVAE underperforms SCAN due to a different reason compared to SCAN_U. JMVAE is able to learn a well structured representation that is almost as disentangled as that of SCAN (see Fig. 17 in Supplementary Materials). However, due to the limitations inherent to learning a joint latent space for the visual and symbolic modalities discussed in Sec. 1, JMVAE does not learn a conceptual abstraction over such disentangled representations. Instead, JMVAE ends up effectively memorising the examples seen during training, which results in the reasonable accuracy yet poor diversity of JMVAE samples over the seen set as shown in Tbl. 1 (see Sec. A.4 in Supplementary Materials for more details), as well as a drop in accuracy between the seen and novel test samples.

Learning recombination operators In this section we evaluate the ability of SCAN to learn logical operators from a small number of examples (100 per each of the three recombination operators). Fig. 8 and Tbl. 1 show qualitative and quantitative results respectively suggesting that SCAN is able to learn the recombination operators and successfully apply them to new concepts. Visual samples generated by SCAN match well the expected ground truth recombined concepts, while maintaining a high degree of variability in the irrelevant visual primitives.

Learning new concepts from symbolic instruction Finally we demonstrate that SCAN can be taught new concepts through purely symbolic instructions. For example, we can test what SCAN has

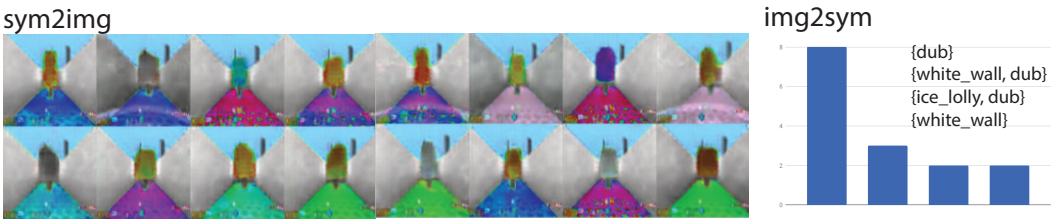

Figure 9: Sym2img and img2sym inferences of SCAN after learning the new concept "dub" through purely symbolic instructions ("dub" IS white_wall AND ice_ lolly). **Sym2img**: visual samples from SCAN when presented with the newly learnt symbol "dub". The samples consistently include ice lollies against white walls, while all other visual factors are randomly sampled from the prior distributions. **Img2sym**: SCAN is presented with an image of a "dub" and is able to describe the image in terms of all concepts it has previously learnt, including the newly acquired concept "dub". The concepts are presented in the order of descending probability of being sampled by SCAN as shown in the histogram.

learnt about a new concept "dub" from an instruction like "dub" IS white_wall AND ice_ lolly. We do so by first asking it to visualise a "dub". Fig. 9 (sym2img) demonstrates that SCAN is able to correctly imagine what a "dub" might look like even though it has never actually seen "dubs" during concept acquisition. The samples are consistent with the instructed meaning of "dub" and show good variability in terms of irrelevant visual primitives. We can also present an image containing an ice lolly presented against a white wall to SCAN and ask it to describe the image. Fig. 9 (img2sym) shows that SCAN is able to use the newly acquired concept "dub" along with the previously acquired concepts "white wall" and "ice lolly" to correctly describe the image.

4 Conclusion

In this paper we described a new approach to learning grounded visual concepts. We first defined concepts as abstractions over structured visual primitives, whereby each concept is grounded in a set of relevant visual factors, while all other (irrelevant) visual factors are defaulted to their prior. We then proposed SCAN, a neural network implementation of such an approach. We showed that unlike the traditional joint multimodal embedding approaches to concept learning, SCAN was able to achieve better performance in terms of accuracy and diversity in a significantly more sample-efficient way. In particular, SCAN was able to discover and learn an implicit hierarchy of abstract concepts from as few as five symbol-image pairs per concept. It was then capable of bi-directional inference, generating diverse image samples from symbolic instructions, and vice versa. The compositional nature of the learnt concepts allowed us to train an extension to SCAN that could perform logical recombination operators. We demonstrated how such operators can be used to traverse the implicit concept hierarchy. In particular, we used symbolic instructions containing recombination operators to teach SCAN the meaning of imaginary new concepts that the model has never experienced visually.

Acknowledgements

We would like to thank Murray Shanahan and Nick Watters for useful feedback that improved the manuscript.

References

Martin Abadi, Ashish Agarwal, and Paul Barham et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *Preliminary White Paper*, 2015.

R. Baillargeon. Young infants' reasoning about the physical and spatial properties of a hidden object. *Cogn. Dev.*, (2):179–200, 1987.

R. Baillargeon. Infants' physical world. *Curr. Dir. Psychol. Sci.*, (13):89–94, 2004.

Charles Beattie, Joel Z. Leibo, Denis Teplyashin, Tom Ward, and Marcus Wainwright et al. Deepmind lab. *arxiv*, 2016.

Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. In *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2013.

Marta Garnelo, Kai Arulkumaran, and Murray Shanahan. Towards deep symbolic reinforcement learning. *arXiv*, 2016.

Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Rezende, and Daan Wierstra. Draw: A recurrent neural network for image generation. In David Blei and Francis Bach (eds.), *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pp. 1462–1471. JMLR Workshop and Conference Proceedings, 2015. URL http://jmlr.org/proceedings/papers/v37/gregor15.pdf.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CVPR*, 2016.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-VAE: Learning basic visual concepts with a constrained variational framework. *ICLR*, 2017a.

Irina Higgins, Arka Pal, Andrei Rusu, Loic Matthey, Christopher Burgess, Alexander Pritzel, Matthew Botvinick, Charles Blundell, and Alexander Lerchner. DARLA: Improving zero-shot transfer in reinforcement learning. *ICML*, 2017b.

Max Jaderberg, Volodymyr Mnih, Wojciech Marian Czarnecki, Tom Schaul, Joel Z Leibo, David Silver, and Koray Kavukcuoglu. Reinforcement learning with unsupervised auxiliary tasks. *ICLR*, 2017.

D. P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv*, 2014.

D. P. Kingma and M. Welling. Auto-encoding variational bayes. *ICLR*, 2014.

Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. *NIPS*, 2014.

Brenden M. Lake, R. Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

Brenden M. Lake, Tomer D. Ullman, Joshua B. Tenenbaum, and Samuel J. Gershman. Building machines that learn and think like people. *arXiv*, 2016.

Loic Matthey, Irina Higgins, Demis Hassabis, and Alexander Lerchner. dsprites: Disentanglement testing sprites dataset. https://github.com/deepmind/dsprites-dataset/, 2017.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *arxiv*, 2013.

Volodymyr Mnih, Koray Kavukcuoglu, David S Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 2015.

Volodymyr Mnih, Adrià Puigdomènech Badia, Mehdi Mirza, Alex Graves, Timothy P. Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. *ICML*, 2016. URL https://arxiv.org/pdf/1602.01783.pdf.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv*, 2016a.

Aaron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray Kavukcuoglu. Conditional image generation with pixelcnn decoders. *NIPS*, 2016b.

G. Pandey and A. Dukkipati. Variational methods for conditional multimodal deep learning. *Intl. J. Conf. on Neural Networks*, 2017.

Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. *CoRR*, abs/1604.07379, 2016. URL http://arxiv.org/abs/1604.07379.

Y. Pu, Z. Gan, R. Henao, X. Yuan, C. Li, A. Stevens, and L. Carin. Variational autoencoder for deep learning of images, labels and captions. *NIPS*, 2016.

S. Reed, Z. Akata, X Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text-to-image. *ICML*, 2016.

Danilo J. Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv*, 2014.

Eleanor H. Rosch. *Cognition and Categorization*, chapter Principles of Categorization, pp. 27–48. Lawrence Erlbaum Associates, Hillsdale, 1978.

David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

K.A. Smith and E. Vul. Sources of uncertainty in intuitive physics. *Top. Cogn. Sci.*, (5):185–199, 2013.

K. Sohn, H. Lee, and X. Yan. Learning structured output representation using deep conditional generative models. *NIPS*, 2015.

E. Spelke. Principles of object perception. *Cognit. Sci.*, (14):25–56, 1990.

Nitish Srivastava and Ruslan Salakhutdinov. Multimodal learning with deep boltzmann machines. *Journal of Machine Learning Research*, 15:2949–2980, 2014.

Masahiro Suzuki, Kotaro Nakayama, and Yutaka Matsuo. Joint multimodal learning with deep generative models. *arxiv*, 2017.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Computer Vision and Pattern Recognition (CVPR)*, 2015. URL `http://arxiv.org/abs/1409.4842`.

Joshua B. Tennenbaum. Bayesian modeling of human concept learning. *NIPS*, 1999.

Ramakrishna Vedantam, Ian Fischer, Jonathan Huang, and Kevin Murphy. Generative models of visually grounded imagination. *arxiv*, 2017.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *NIPS*, 2010.

Weiran Wang, Xinchen Yan, Honglak Lee, and Karen Livescu. Deep variational canonical correlation analysis. *arxiv*, 2016.

Wikipedia. HSL and HSV. `https://en.wikipedia.org/wiki/HSL_and_HSV/`, 2017. [Online; accessed 15-June-2017].

X. Yan, J. Yang, K. Sohn, and H. Lee. Attribute2image: Conditional image generation from visual attributes. *ECCV*, 2016.

A Supplementary Information

A.1 Model details

β-VAE We re-used the architecture and the training setup for β-VAE specified in Higgins et al. (2017b). In particular, we used L2 loss within a pre-trained denoising autoencoder (DAE) to calculate the reconstruction part of the β-VAE loss function. The DAE was trained with occlusion-style masking noise in the vein of (Pathak et al., 2016), with the aim for the DAE to learn a semantic representation of the input frames. Concretely, two values were independently sampled from $U[0, W]$ and two from $U[0, H]$ where W and H were the width and height of the input frames. These four values determined the corners of the rectangular mask applied; all pixels that fell within the mask were set to zero.

The DAE architecture consisted of four convolutional layers, each with kernel size 4 and stride 2 in both the height and width dimensions. The number of filters learnt for each layer was {32, 32, 64, 64} respectively. The bottleneck layer consisted of a fully connected layer of size 100 neurons. This was followed by four deconvolutional layers, again with kernel sizes 4, strides 2, and {64, 64, 32, 32} filters. The padding algorithm used was 'SAME' in TensorFlow (Abadi et al., 2015). ELU non-linearities were used throughout. The optimiser used was Adam (Kingma & Ba, 2014) with a learning rate of 1e-3 and ϵ=1e-8. We pre-trained the DAE for 200,000 steps, using batch size of 100 before training β-VAE.

β-VAE architecture was the following. We used an encoder of four convolutional layers, each with kernel size 4, and stride 2 in the height and width dimensions. The number of filters learnt for each layer was {32, 32, 64, 64} respectively. This was followed by a fully connected layer of size 256 neurons. The latent layer comprised 64 neurons parametrising 32 (marginally) independent Gaussian distributions. The decoder architecture was simply the reverse of the encoder, utilising deconvolutional layers. The decoder used was Bernoulli. The padding algorithm used was 'SAME' in TensorFlow. ReLU non-linearities were used throughout. The reconstruction error was taking in the last layer of the DAE (in the pixel space of DAE reconstructions) using L2 loss and before the non-linearity. The optimiser used was Adam with a learning rate of 1e-4 and ϵ=1e-8. We pre-trained β-VAE until convergence using batch size of 100. The disentangled β-VAE had $\beta = 53$, while the unstructured baseline had $\beta = 0.02$.

SCAN The encoder and decoder of SCAN were simple single layer MLPs with 100 hidden units and ReLU non-linearities. The decoder was parametrised as a Bernoulli distribution over the output space of size 375. We set $\beta_y = 1, \lambda = 10$ for all our experiments. We trained the model using Adam optimiser with learning rate of 1e-4 and batch size 16.

SCAN recombination operator The recombination operator was implemented as a convolutional operator with kernel size 1 and stride 1. No non-linearity was used, and the operator output the parameters of each component of the diagonal multivariate Gaussian distribution of the recombined concept as a linear combination of the parameters of the corresponding components of the two input concepts. The optimizer is Adam with a learning rate of 1e-3 and batch size 100 was used. We found that the recombination operator converged with 10k steps.

JMVAE The JMVAE was trained using the loss as described in (Suzuki et al., 2017):

$$\begin{aligned}\mathcal{L}_{JM}(\theta_x, \theta_y, \phi_x, \phi_y, \phi; \mathbf{x}, \mathbf{y}, \alpha) =& \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}\left[\log p_{\theta_x}(\mathbf{x}|\mathbf{z})\right] \; + \; \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}\left[\log p_{\theta_y}(\mathbf{y}|\mathbf{z})\right] \\ & - D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})||p(\mathbf{z})\right) \\ & - \alpha\big[D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})||q_{\phi_x}(\mathbf{z}|\mathbf{x})\right) \\ & \quad + D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})||q_{\phi_y}(\mathbf{z}|\mathbf{y})\right)\big] \end{aligned} \tag{6}$$

Where α was a parameter that was tuned over from the range {0.01, 0.1, 1.0, 10.0} as in the original paper. We found that best results were obtained for $\alpha = 1.0$, which constituted the model that we reported results on.

The architectural choices were made to match as closely as possible as those made for SCAN. Thus the visual encoder q_{ϕ_x} consisted of four convolutional layers, each with kernel size 4 and stride 2 in both the height and width dimensions, with {32, 32, 64, 64} filters learned at the respective layers. This was then appended with a fully connected network with hidden size 256 units, with a final output dimensionality of 64 (corresponding to the parameters of a 32-dimensional diagonal Gaussian latent distribution). The symbol encoder q_{ϕ_y} consisted of a single layer MLP with 100 hidden units, as in SCAN. The joint encoder q_ϕ consisted of the same convolutional stack as in the visual encoder to process the visual input, while the symbol input was passed through a two-layer MLP with 32 and 100 hidden units. These two embeddings were then concatenated and passed through a further two-layer MLP of 256 hidden units each, before outputting the 64 parameters of the diagonal Gaussian latents.

The visual decoder p_{θ_x} was simply the reverse of the visual encoder, using transposed convolutions instead of convolutions. Similarly, the symbol decoder p_{θ_x} was again a single layer, 100 hidden unit MLP. The output distributions of both decoders were parameterised as Bernoullis. The model was trained using the Adam optimiser with a learning rate of 1e-4 and a batch size of 16.

Accuracy and diversity evaluation The classifier used to evaluate the samples generated by each model was trained to discriminate the four room configuration factors in the DeepMind Lab dataset. We used a network of four 2-strided deconvolutional layers (with filters in each successive layer of {32, 64, 128, 256}, and kernels sized 3x3), followed by a fully connected layer with 256 neurons, with ReLU activations used throughout. The output layer consisted of four fully connected softmax heads, one for each predicted factor (with dimensionality 16 for each of the colour factors, 3 for object identity). The classifier was trained until convergence using the Adam optimiser, with a learning rate of 1e-4 and batch size of 100 (reaching an overall accuracy of 0.991975).

Overall accuracy was computed as the average top-k accuracy across the factors (with $k = 3$ for the colour factors, and $k = 1$ for the object identity factor), against the ground-truth factors specified by the concept used to generate each sample. The classifier was used to compute the sym2img top-k of the factors in each image sample.

Sample diversity of the sym2img data was characterised by estimating the KL divergence of the irrelevant factor distribution inferred for each concept with a flat distribution, $KL(u(\mathbf{y}_i)||p(\mathbf{y}_i))$. Here, $p(\mathbf{y}_i)$ is the joint distribution of the irrelevant factors in the sym2img set of images generated from the ith concept, which we estimated by averaging the classifier predictions across those images. $u(\mathbf{y}_i)$ is the desired (flat) joint distribution of the same factors (i.e., where each factor value has equal probability). We also computed the expected KL if $p(\mathbf{y}_i)$ were estimated using the samples drawn from the flat distribution $u(\mathbf{y}_i)$. We report the mean of this KL across all the k-grams.

A.2 Training details

Learning grounded concepts Our DeepMind Lab (Beattie et al., 2016) dataset contained 73 frames per room, where the configuration of each room was randomly sampled from the outer product of the four data generative factors: object identity and colour, wall and floor colour (12,288 unique factor combinations). All models were trained using a randomly sampled subset of 100 of these concepts and their corresponding randomly sampled frames.

The majority of factors to be learnt in this dataset correspond to colours (floor, wall and object). We found that it was hard to learn disentangled representations of these data generative factors with β-VAE. We believe this was the case because β-VAE requires a degree of smoothness in pixel space when traversing a manifold for a particular data generative factor in order to correctly learn this factor (Higgins et al., 2017a). The intuitively smooth notion of colour, however, is disrupted in RGB space (see Fig. 10). Instead, the intuitive human notion of colour is more closely aligned with hue in HSV space. Hence we added a pre-processing step that projected the DeepMind Lab frames from RGB to HSV space before training β-VAE. We found that this enabled β-VAE to achieve good disentangling results.

Figure 10: **A**: Comparison of hue traversal in HSV space, which closely aligns with the intuitive human understanding of colour, and the equivalent highly non-monotonic changes in RGB space. *H* stands for hue, *S* stands for saturation and *V* stands for value/brightness. Adapted from (Wikipedia, 2017). **B**: Visualisation of colours used in DeepMind Lab in RGB. **C**: Visualisation of colours used in DeepMind Lab in HSV. It can be seen that the HSV projection of the DeepMind Lab colours appears significantly more structured than the equivalent RGB projection.

Learning recombination operators The recombination operator was trained by sampling two concepts, $\mathbf{y}_1$ and $\mathbf{y}_2$, and an operator $\mathbf{h}$ as input. The output was provided by inferring the ground truth recombined concept $\mathbf{z}_r$ from an image $\mathbf{x}_i$. The ground truth image $\mathbf{x}_i$ was obtained by applying binary logical operations

corresponding to $\mathbf{h}$ to binary symbols $\mathbf{y}_1$ and $\mathbf{y}_2$. This produces the ground truth recombined symbol $\mathbf{y}_r$, which can then be used to sample a corresponding ground truth image $\mathbf{x}_r$ from the dataset.

To make sure that the logical operators were not presented with nonsensical instructions, we followed the following logic for sampling the inputs and outputs during training. We always trained the UNION operator on two unique unigrams. The INTERSECT operator was trained by sampling two k-grams with $k \in \{2, 3, 4\}$ and a restriction that the intersection cannot be and empty set. The ignore operator was trained by sampling a k-gram with $k \in \{2, 3, 4\}$ and a unigram selected from one of the factors specified by the k-gram.

Fast learning of new concepts from symbolic instructions In order to avoid catastrophic forgetting of previously learnt symbol-to-concept associations within SCAN_{IS}, we augment the dataset of new concepts with examples of old concepts previously learnt from visual examples (e.g. "ice_lolly" or "white_wall"). In order to make the training set homogenous (where all concepts, old and new, are explained in terms of symbolic instructions that include a logical operator), we use symbolic instructions to describe old concepts (e.g. {"white_wall" IS "white_wall" AND "red_floor"}, where the second concept is randomly sampled to create a diverse yet sensible set of instructions defining "white_wall"). The parameters of SCAN_{IS} get copied across to the corresponding parameters of SCAN_y ($\phi_{\text{IS}} \rightarrow \phi_y$, $\theta_{\text{IS}} \rightarrow \theta_y$) once learning of SCAN_{IS} converges.

A.3 dSprites experiments

In this section we describe additional experiments testing SCAN on the dSprites (Matthey et al., 2017) dataset.

A.3.1 Dataset and Task

dSprites dataset (see Fig. 11 for example frames) was chosen for its simplicity, full ground truth specification and the fact that β-VAE is known to learn good disentangled representations on this dataset (Higgins et al., 2017a). Hence we use dSprites for proof of principle evaluation of the SCAN framework. The dataset consists of binary sprites fully specified by five ground truth factors: position x (32 values), position y (32 values), scale (6 values), rotation (40 values) and sprite identity (3 values). Our experiments investigated whether SCAN could discover and learn the implicit hierarchy of concepts of the dataset. In practice we quantised the space spanned by three of the data generative factors - horizontal and vertical positions, and scale - into halves, and assigned one-hot encoded symbols to each of the $\sum_{k=1}^{K} \binom{K}{k} N^k = 26$ possible concepts to be learnt (since $K = 3$ is the number of factors to be learnt and $N = 2$ is the number of values each factor can take).

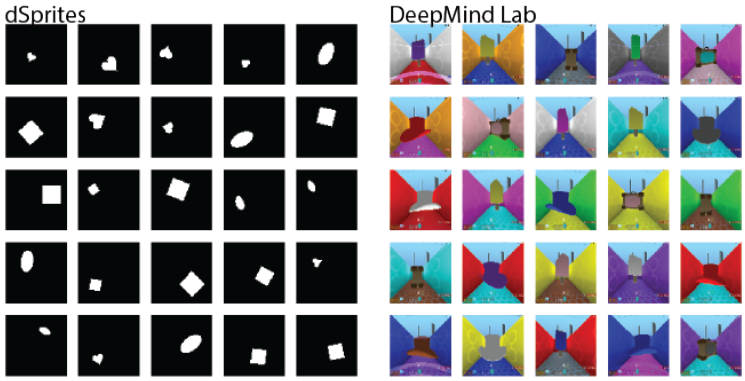

Figure 11: Example frames from dSprites (Matthey et al., 2017) and DeepMind Lab (Beattie et al., 2016) datasets.

A.3.2 Experiments

In this section we use a simple dSprites dataset to demonstrate a proof of principle of our proposed SCAN framework. In particular we show the following:

1. SCAN can discover and learnt the implicit hierarchy of compositional visual concepts from a small number of examples (as few as 20 example images per concept)
2. Such fast learning only works when concepts are grounded in structured rather than unstructured visual representations
3. When a concept is learnt, it becomes an abstraction over the space of disentangled visual building blocks. In particular, those visual building blocks that are specified by the concept get low inferred variances, while all other visual building blocks default to the wide prior distribution

4. A fully trained SCAN is capable of generating diverse visual samples from symbol inputs

Concept learning We argue that SCAN framework requires structured visual building blocks in order to learn a grounded abstract concept hierarchy. In the following set of experiments we therefore compare the performance of SCAN coupled either with a model that learnt to infer structured disentangled representations of the visual factors of variation (β-VAE with $\beta = 12$) or a model that learnt an unstructured visual embedding implemented by training a β-VAE with $\beta = 0$. We chose to use a β-VAE with $\beta = 0$ rather than a deterministic autoencoder to keep the parametrised structure of the latent space necessary for SCAN to operate. Fig. 12A demonstrates that both versions of β-VAE ($\beta = 12$ and $\beta = 0$) learnt to reconstruct the data well. It has previously been shown that β-VAE with $\beta > 1$ learns to infer well disentangled representations of the data generative factors on the dSprites dataset (Higgins et al., 2017a).

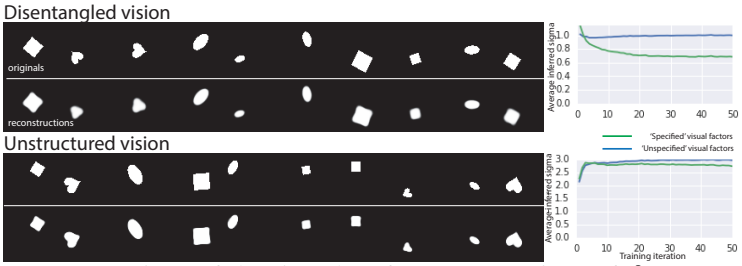

Figure 12: **A**: Reconstructions of dSprites data from two versions of β-VAE: one that learnt a structured disentangled latent representation ($\beta = 12$), and one that learnt an unstructured distributed latent representation ($\beta = 0$). **B**: Changes in average inferred variances for specified and unspecified latents $\mathbf{z}_y$ of the two versions of SCAN during training. The distinction between specified and unspecified latents in the SCAN with unstructured vision was done by tracking the highest and lowest 30% inferred sigmas.

We train SCAN on a random subset of image-symbol pairs (x_i, y_i). Each image x_i in the dSprites dataset can be described in terms of 7 possible concepts and their corresponding symbols y_i, which results in the total of $737,280 * 7 = 5,160,960$ possible (x_i, y_i) pairs. The training subset consists of 20 randomly sampled example images for each of the 26 possible concepts, and hence is $< 0.01\%$ of the full dataset.

As mentioned in Sec. 2.1, a concept is considered learnt and grounded in vision if it specifies narrow distributions over the relevant visual building blocks and wide (prior) distributions over all other visual building blocks. We tested the changes in inferred variances within the latent conceptual space $\mathbf{z}_y$ of SCAN for the *relevant* and *irrelevant* visual primitives during training. Fig. 12B compares such curves between a SCAN grounded in a structured visual representation (right top) and an unstructured visual representation (right bottom). It can be seen that on average over a sample of random concepts SCAN coupled with structured vision shows a wide separation in inferred variances for the *relevant* and *irrelevant* factors. As expected, the *irrelevant* factors default to the unit Gaussian prior variance ($\sigma \approx 1$), while the *relevant* factors average around $\sigma \approx 0.6$. Such separation is missing for the SCAN coupled with unstructured vision.

Concept understanding Since dSprites is a binary dataset, we can quantify how well SCAN understands the meaning of different concepts $\mathbf{z}_y$ from symbolic instructions $\mathbf{y}_i$ by generating a batch of visual samples $\hat{\mathbf{x}} \sim p_{\theta_x}(\mathbf{x}|\mathbf{z}_y)$ for each concept and counting the number of white pixels within each of the four quadrants of the canvas (for location) or in total in the whole image (for scale). This can be compared to similar values calculated over a batch of ground truth images that match the same input symbols $\mathbf{y}_i$. Samples from SCAN coupled with structured vision matched closely the statistics of the ground truth samples (see Fig. 13). SCAN coupled with unstructured vision, however, failed to produce meaningful samples despite being able to reconstruct the dataset almost perfectly (Fig. 12). It is interesting to note the diversity of the samples produced by SCAN coupled with structured vision. For example, when one contrasts the diversity of scale in the SCAN samples when presented with a concept "left" compared to "large top", the latter produces less scale variability as expected.

A.4 DeepMind Lab experiments

SCAN with unstructured vision We first demonstrate that SCAN with unstructured vision was based on a β-VAE that learnt a good (yet entangled) representation of the DeepMind Lab dataset. Fig. 14 shows that such an entangled β-VAE was able to reconstruct the data well, however due to the entangled nature of its learnt representations, latent traversal plots and samples are not as good as those of a disentangled β-VAE (Fig. 5).

Due to the unstructured nature of the visual latent space, the additional KL terms of SCAN loss function (Eq. 3) is not able to pick out the subsets of relevant and irrelevant visual primitives corresponding to the concept to be

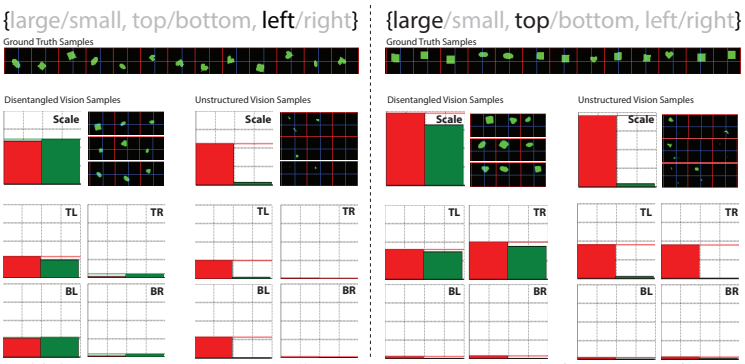

Figure 13: Quantifying concept understanding. Symbol to image inference performance comparison of SCAN trained with disentangled vision and unstructured vision. Inference is drawn from two example symbols - "left" (left) and "large top" (right). First line in each subplot demonstrates ground truth samples from dSprites dataset that correspond to the respective symbol. Next line illustrates the comparative performance of disentangled SCAN vs the unstructured baseline. Rightwards plots in the third line demonstrate respective image samples drawn from the models. The bar plots compare the performance of SCAN (green) vs ground truth (red). The bar plots in the third row (Scale) count the average number of pixels present in samples, and are indicative of scale concepts. The bar plots in rows four and five count the average number of pixels present in each quadrant of the canvas (T - top, B - bottom, R - right, L - left). This is indicative of understanding positional concepts. The closer green bars are to the ground truth red bars, the better the model's understanding of the learnt concepts.

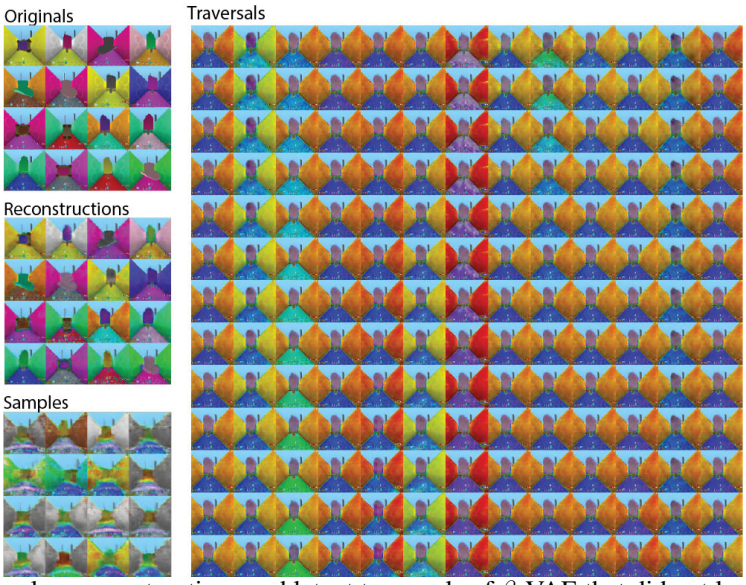

Figure 14: Samples, reconstructions and latent traversals of β-VAE that did not learn a structured disentangled representation ($\beta = 0.02$). It is evident that the model learnt to reconstruct the data despite learning an entangled latent space.

learnt. This is because concepts refer to a particular set of data generative factors. These data generative factors, however, affect the values of numerous latent units of the entangled β-VAE. Hence almost all of the latents $\mathbf{z}_x$ of β-VAE show a lot of variability when β-VAE is presented with different visual examples of a particular concept. Due to the properties of the $KL(\mathbf{z}_x||\mathbf{z}_y)$ described in Sec. 2.3, SCAN struggles to pick out the specified vs unspecified visual conceptual attributes (compared Fig. 15 to Fig. 7B). This then disrupts the ability of SCAN with unstructured vision to learn useful concepts, as demonstrated in Fig. 16.

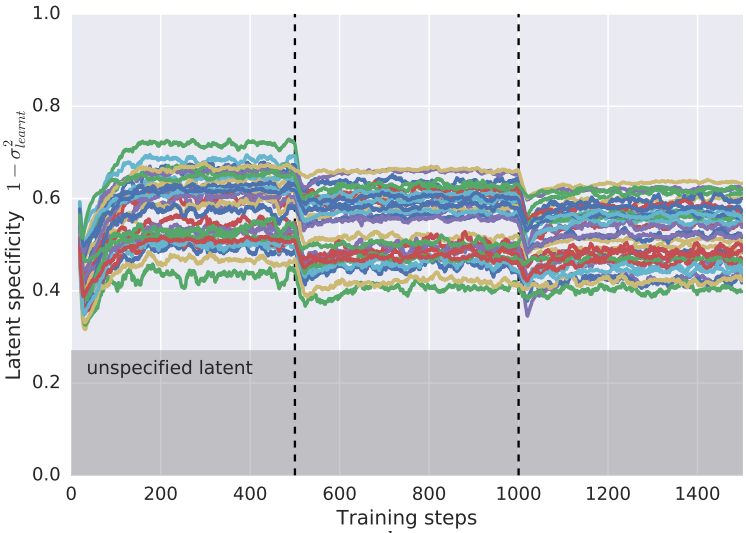

Figure 15: Average specificity of inferred latents z_y^k of SCAN with a unstructured vision during the "cyan wall" concept learning curriculum task described in Sec. 3. While SCAN with disentangled vision progressively refines the set of visual primitives that it assigns to the concept of wall colour as it experiences more diverse visual examples at step 500 and 1000 (Fig. 7), SCAN with unstructured vision never manages to do so.

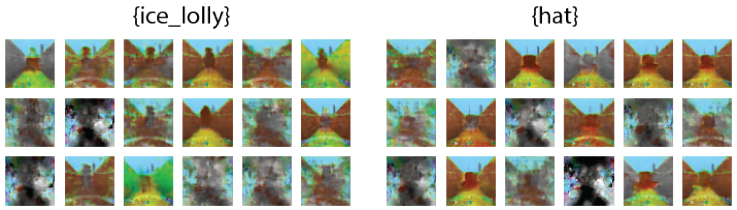

Figure 16: Visual samples (sym2img) of SCAN grounded in unstructured vision when presented in concepts "hat" and "ice_lolly". It is evident that the model struggled to learn a good understanding of the meaning of these concepts.

A.4.1 JMVAE

In this section we provide some insights into the nature of representations learnt by JMVAE (Suzuki et al., 2017). Fig. 17 demonstrates that after training JMVAE is capable of reconstructing the data and drawing reasonable visual samples. Furthermore, the latent traversal plots indicate that the model learnt a reasonably disentangled representation of the data generative factors. Apart from failing to learn a latent to represent the spawn animation and a latent to represent all object identities (while the hat and the ice lolly are represented, the suitcase is missing), the representations learnt by JMVAE match those learnt by β-VAE (compare Figs. 5 and 17). Note, however, that unlike β-VAE that managed to discover and learn a disentangled representation of the data generative factors in a completely unsupervised manner, JMVAE was able to achieve its disentangling performance by exploiting the extra supervision signal coming from the symbolic inputs.

Due to the inherent information inequality between the visual and symbolic domains, JMVAE is unable to learn a hierarchical compositional latent representation of concepts like SCAN does. Instead, it learns a flat representation of visual primitives like the representation learnt by β-VAE. Such a flat representation is problematic, as evidenced by the accuracy/diversity metrics shown in Tbl. 1. Further evidence comes from examining the sym2img samples produced by JMVAE (see Fig. 18). It can be seen that JMVAE fails to learn the abstract concepts as defined in Sec. 2.1. While the samples in Fig. 18 mostly include correct wall colours that match their respective input symbols, the samples have limited diversity. Many samples are exact copies of each other - a sign of mode collapse.

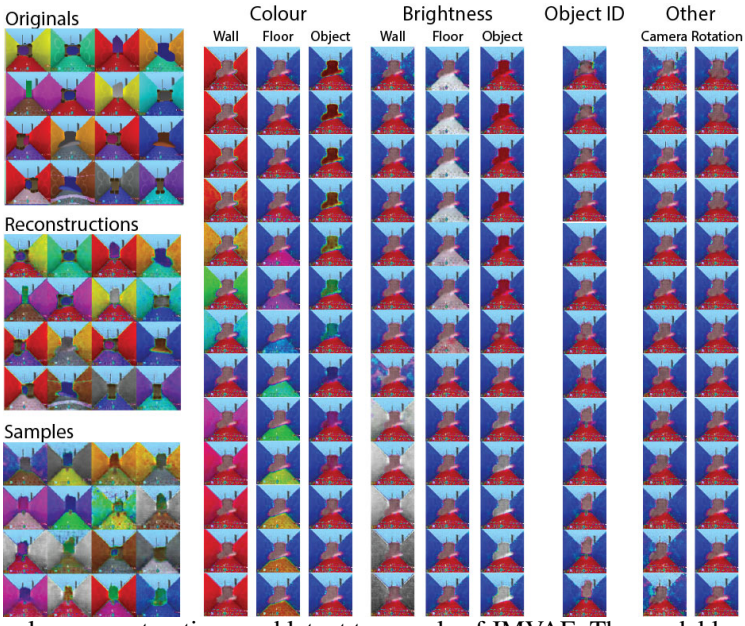

Figure 17: Samples, reconstructions and latent traversals of JMVAE. The model learns good disentangled latents, making use of the supervised symbolic information available.

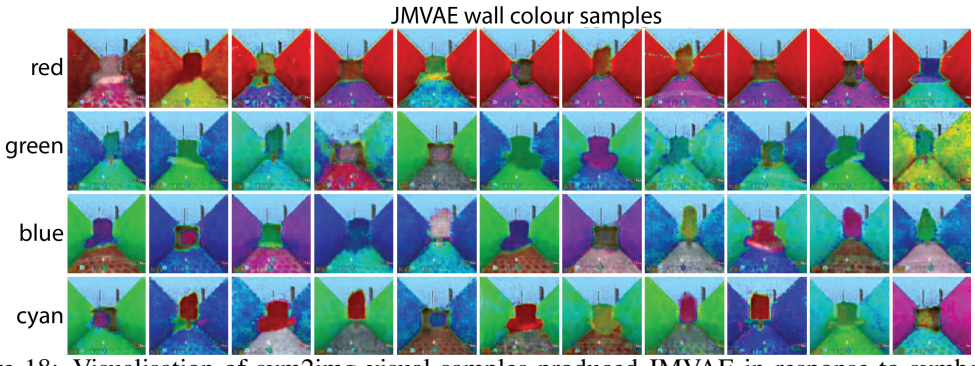

Figure 18: Visualisation of sym2img visual samples produced JMVAE in response to symbols specifying wall colour names. It is evident that the model suffers from mode collapse, since a significant number of samples are copies of each other.