

---

# Systematic Visual Reasoning through Object-Centric Relational Abstraction

---

**Taylor W. Webb\***

Department of Psychology  
University of California, Los Angeles  
Los Angeles, CA  
taylor.w.webb@gmail.com

**Shanka Subhra Mondal\***

Department of Electrical and Computer Engineering  
Princeton University  
Princeton, NJ  
smondal@princeton.edu

**Jonathan D. Cohen**

Princeton Neuroscience Institute  
Princeton University  
Princeton, NJ  
jdc@princeton.edu

**\* Equal Contribution**

## Abstract

Human visual reasoning is characterized by an ability to identify abstract patterns from only a small number of examples, and to systematically generalize those patterns to novel inputs. This capacity depends in large part on our ability to represent complex visual inputs in terms of both objects and relations. Recent work in computer vision has introduced models with the capacity to extract object-centric representations, leading to the ability to process multi-object visual inputs, but falling short of the systematic generalization displayed by human reasoning. Other recent models have employed inductive biases for relational abstraction to achieve systematic generalization of learned abstract rules, but have generally assumed the presence of object-focused inputs. Here, we combine these two approaches, introducing Object-Centric Relational Abstraction (OCRA), a model that extracts explicit representations of both objects and abstract relations, and achieves strong systematic generalization in tasks involving complex visual displays.

## 1 Introduction

When presented with a visual scene, human reasoners have a capacity to identify not only the objects in that scene, but also the relations between objects, and the higher-order patterns formed by multiple relations. These abilities are fundamental to human visual intelligence, enabling the rapid induction of abstract rules from a small number of examples, and the systematic generalization of those rules to novel inputs [27, 20, 11, 22]. For example, when presented with the image in Figure 1, one can easily determine that the objects in the top and bottom rows are both governed by a common abstract pattern (ABA). Furthermore, one could easily then generalize this pattern to *any* potential shapes, even those that one has never observed before. Indeed, this is arguably what it means for a pattern to be considered *abstract*.

Though this task appears effortless from the perspective of human reasoning, it exemplifies the type of problems that pose a significant challenge for standard neural network methods. Rather than inducing an abstract representation of the underlying rules in a given task, neural networks tend to

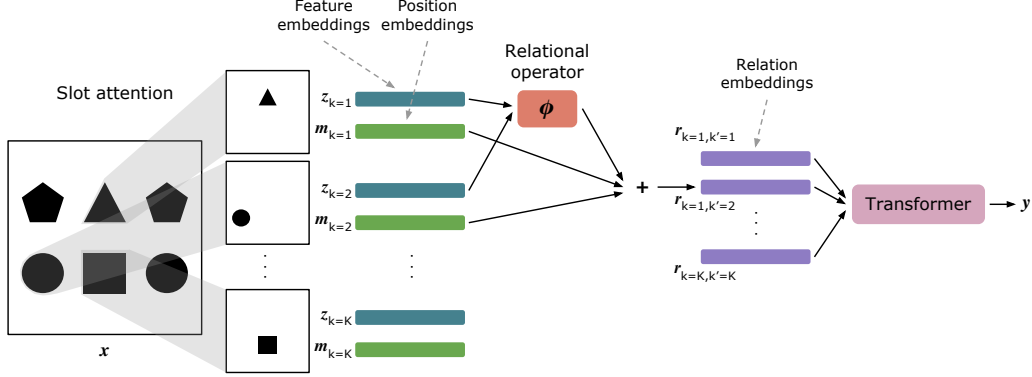


Figure 1: **Object-Centric Relational Abstraction (OCRA)**. OCRA consists of three core components, responsible for modeling objects, relations, and higher-order relations. First, given a visual input  $x$ , slot attention extracts object-centric representations, consisting of factorized feature embeddings  $z_{k=1\dots K}$  and position embeddings  $m_{k=1\dots K}$ . Second, pairwise relation embeddings  $r_{kk'}$  are computed by first passing each pair of feature embeddings to a relational operator  $\phi$ , and then additively incorporating their position embeddings. This allows visual space to serve as an indexing mechanism for visual relations. Finally, relational embeddings are passed to a transformer to model the higher-order patterns formed by multiple relations.

overfit to the specific details of the examples observed during training, thus failing to generalize those rules to novel inputs [21, 3, 28].

To address this, a number of neural network methods have recently been proposed that incorporate strong inductive biases for *relational abstraction* [44, 45, 17, 1], also referred to as a ‘relational bottleneck’. The key insight underlying these approaches is that abstract visual patterns are often characterized by the *relations* between objects, rather than the features of the objects themselves. For example, the pattern seen in Figure 1 is characterized by the relation between the shape and position of the objects. By biasing architectures to process visual inputs in terms of relations between objects, these recent approaches have achieved strong systematic (i.e., out-of-distribution) generalization of learned abstract rules, given only a small number of training examples.

However, a major assumption of these approaches is that relations are computed between *objects*. Accordingly, these methods have generally relied on inputs consisting of pre-segmented visual objects, rather than multi-object scenes. Furthermore, these methods can sometimes be brittle to task-irrelevant visual variation, such as spatial jitter or Gaussian noise [38]. It is thus unclear whether and how these approaches might scale to complex and varied real-world settings.

In this work, we take a step toward the development of relational reasoning algorithms that can operate over more realistic visual inputs. To do so, we turn to object-centric representation learning [13, 6, 23, 10], a recently developed approach in which visual scenes are decomposed into a discrete set of object-focused latent representations, without the need for veridical segmentation data. Though this approach has shown success on a number of challenging visual tasks [9, 34, 25], it has not yet been integrated with the strong relational inductive biases that have previously been shown to enable systematic, human-like generalization [44, 45, 17]. Here, we propose Object-Centric Relational Abstraction (OCRA), a model that combines the strengths of these two approaches. Across a suite of visual reasoning tasks, we find that OCRA is capable of systematically generalizing learned abstract rules from complex, multi-object visual displays, significantly outperforming a number of competitive baselines in most settings.

## 2 Approach

Figure 1 shows a schematic depiction of our approach. OCRA consists of three core components, each responsible for a different element of the reasoning process. These three elements can be summarized as 1) extraction of object-centric representations, 2) computation of pairwise relational embeddings,

and 3) processing of higher-order relational structures – patterns across multiple relations – in order to identify abstract rules. We describe each of these three components in the following sections.

## 2.1 Object-centric representation learning

OCRA employs slot attention [23] to extract object-centric representations from complex, multi-object visual inputs. Given an image  $x$ , the goal of slot attention is to extract a set of latent embeddings (i.e., *slots*), each of which captures a focused representation of a single visual object, typically without access to ground truth segmentation data.

To do so, the image is first passed through a series of convolutional layers, generating a feature map  $\mathbf{feat} \in \mathbb{R}^{H \times W \times D}$ . Positional embeddings  $\mathbf{pos} \in \mathbb{R}^{H \times W \times D}$  are then generated by passing a 4d position code (encoding the four cardinal directions) through a separate linear projection. Feature and position embeddings are additively combined, and then passed through a series of position-wise fully-connected layers. This is then flattened, yielding  $\mathbf{inputs} \in \mathbb{R}^{N \times D}$  (where  $N = H \times W$ ), over which slot attention is performed.

The set of  $K$  *slots*  $\in \mathbb{R}^{K \times D}$  is then randomly initialized (from a shared distribution with learned parameters), and a form of transformer-style cross-attention is employed to attend over the pixels of the feature map. Specifically, each slot emits a query  $\mathbf{q}(\mathbf{slots}) \in \mathbb{R}^{K \times D}$  (through a linear projection), and each location in the feature map emits a key  $\mathbf{k}(\mathbf{inputs}) \in \mathbb{R}^{N \times D}$  and a value  $\mathbf{v}(\mathbf{inputs}) \in \mathbb{R}^{N \times D}$ . For each slot, a query-key attention operation is then used to generate an attention distribution over the feature map  $\mathbf{attn} = \text{softmax}(\frac{1}{\sqrt{D}} \mathbf{k}(\mathbf{inputs}) \cdot \mathbf{q}(\mathbf{slots})^\top)$  and a weighted mean of the values  $\mathbf{updates} = \mathbf{attn} \cdot \mathbf{v}(\mathbf{inputs})$  is used to update the slot representations using a Gated Recurrent Unit [7], followed by a residual MLP. This process is repeated for  $T$  iterations, leading to a competition between the slots to account for each pixel in the feature map.

For the purposes of computing relational embeddings, we employed a factorized object representation format, in which position is represented distinctly from other object features. For a given input image, after performing  $T$  iterations of slot attention, we used the final attention map  $\mathbf{attn}_T \in \mathbb{R}^{K \times N}$  to compute feature- and position-specific embeddings for each slot:

$$\mathbf{z}_{k=1..K} = \mathbf{attn}_T \cdot \text{flatten}(\mathbf{feat}) \quad (1)$$

$$\mathbf{m}_{k=1..K} = \mathbf{attn}_T \cdot \text{flatten}(\mathbf{pos}) \quad (2)$$

where  $\mathbf{feat}$  and  $\mathbf{pos}$  represent the feature- and position-embedding maps (prior to being combined). As will be seen in the following section, this factorized representation allows visual space to serve as an indexing mechanism for pairwise relations, mirroring the central role of spatial organization in human visual reasoning [24].

Although slot attention can in principle be learned end-to-end in the service of a specific downstream reasoning task, here we chose to pre-train slot attention using an unsupervised objective. This decision reflects the assumption that human-like visual reasoning does not occur in a vacuum, and visual representations are generally not learned from scratch when performing each new task. Instead, visual object representations are shaped by a wealth of non-task-specific prior experience. To model the effects of this prior experience, we pre-trained slot attention to reconstruct randomly generated multi-object displays, using a slot-based autoencoder framework (more details in Section 4.3.1). Importantly, for experiments testing generalization of abstract rules to novel objects, we ensured that these objects did not appear in the pre-training data for slot attention.

## 2.2 Relational embeddings

After extracting object embeddings, we employed a novel relational embedding method, designed to allow OCRA to abstract over the perceptual features of individual objects (and thus systematically generalize learned rules to inputs with novel perceptual features). This embedding method consisted of two steps. First, we applied a relational operator  $\phi$  to compute the pairwise relation between two feature embeddings  $\mathbf{z}_k$  and  $\mathbf{z}_{k'}$ . In the present work, we define this operator as:

$$\phi(\mathbf{z}_k, \mathbf{z}_{k'}) = (\mathbf{z}_k \mathbf{W}_z \cdot \mathbf{z}_{k'} \mathbf{W}_z) \mathbf{W}_r \quad (3)$$

where  $\mathbf{z}_k$  and  $\mathbf{z}_{k'}$  are first projected through a shared set of linear weights  $\mathbf{W}_z \in \mathbb{R}^{D \times D}$ , and the dot product between these weighted embeddings is then projected back out into the original dimensionality using another set of weights  $\mathbf{W}_r \in \mathbb{R}^{1 \times D}$ . Using dot products to model pairwise relations introduces a layer of abstraction between the perceptual features of the objects and the downstream reasoning process, in that the dot product *only* represents the relations between objects, rather than those objects' individual features. We hypothesized that this relational bottleneck would lead to improved generalization to inputs with novel perceptual features. Additionally, the shared projection weights ( $\mathbf{W}_z$ ) enable the model to learn which features are relevant for a given task, and to only compute relations between those features.

After applying the relational operator, we then compute the relational embedding for objects  $k$  and  $k'$  as:

$$\mathbf{r}_{kk'} = \phi(\mathbf{z}_k, \mathbf{z}_{k'}) + \mathbf{m}_k \mathbf{W}_m + \mathbf{m}_{k'} \mathbf{W}_m \quad (4)$$

where the position embeddings  $\mathbf{m}_k$  and  $\mathbf{m}_{k'}$  are each projected through a shared set of linear weights  $\mathbf{W}_m \in \mathbb{R}^{D \times D}$  and added to the output of  $\phi$ . This allows the relation between objects  $k$  and  $k'$  to be indexed by their spatial position, thus endowing OCRA with an explicit variable-binding mechanism.

### 2.3 Processing of higher-order relations

After computing all  $K^2$  pairwise relational embeddings, we then pass these embeddings to a reasoning process to extract higher-order relations – patterns formed between multiple relations. This is an essential component of visual reasoning, as abstract rules, such as the one displayed in Figure 1, are typically defined in terms of higher-order relations. We model these higher-order interactions using a transformer [40]:

$$\mathbf{y} = \text{Transformer}(\mathbf{r}_{k=1, k'=1} \dots \mathbf{r}_{k=K, k'=K}) \quad (5)$$

where the inputs are formed by the set of all pairwise relational embeddings  $\mathbf{r}_{kk'}$  (for all  $k = 1 \dots K$  and all  $k' = 1 \dots K$ ), and  $\mathbf{y}$  is a task-specific output. Because we employ a symmetric relational operator in the present work, we do not in practice present the full matrix of pairwise comparisons, but instead limit these comparisons to those in the upper triangular matrix.

## 3 Related Work

A number of methods for annotation-free object segmentation have recently been proposed [13, 6, 23, 10], including the slot attention method employed by our proposed model [23]. Though the details of these methods differ, they share the general goal of decomposing visual scenes into a set of latent variables corresponding to objects, without access to ground truth segmentation data. Building on the success of these object-encoding methods, some recent work has developed object-centric approaches to visual reasoning that combine slot-based object representations with transformer-based architectures [9, 34, 25]. This work has yielded impressive performance on challenging computer vision tasks, such as question answering from video [9], or complex multi-object visual analogy problems [25], but has thus far been limited to generalization to similar problem types following extensive task-specific training.

In the present work, our goal was to develop a model capable of human-like systematic generalization in abstract reasoning tasks. Our proposed approach builds on the insights of other recent work in this area, most notably including the *Emergent Symbol Binding Network (ESBN)* [45] and *CoRelNet* [17] architectures, both of which employed the concept of a relational bottleneck in different ways. In the ESBN architecture, separate control and perception pathways interact indirectly through a key-value memory, allowing the control pathway to learn representations that are abstracted over the details represented in the perceptual pathway. Kerg et al. [17] proposed that the ESBN's capacity for abstraction was due to the implicit encoding of relational information in its memory operations, which are mediated by a dot-product-based retrieval operation. Kerg et al. also proposed a novel

architecture, CoRelNet, that makes this inductive bias more explicit. The CoRelNet architecture operates by explicitly computing the full matrix of relations (based on dot products) between each pair of objects in its input, and then passing these relations to an MLP for further processing. Both ESBN and CoRelNet displayed strong systematic generalization of learned abstract rules from a small set of examples. Importantly, however, both of these architectures depended on the presence of pre-segmented objects as input. A primary contribution of the present work is to combine these inductive biases for relational abstraction (i.e. relational bottleneck) with object-centric encoding mechanisms, and to introduce the use of a spatial indexing mechanism that allows the model to apply these inductive biases to spatially embedded, multi-object visual inputs.

In addition to lacking object encoding mechanisms, it was recently shown that ESBN can sometimes be overly brittle to task-irrelevant visual variation. Vaishnav and Serre [38] evaluated ESBN and other reasoning architectures on abstract visual reasoning problems similar to those evaluated by Webb et al. [45], but with the introduction of either spatial jitter or Gaussian noise, finding that this significantly impaired performance in the ESBN (as well as other reasoning architectures, such as the transformer [40] and relation net [30]). Vaishnav and Serre also introduced a novel architecture, GAMR, that was more robust to these sources of noise, and able to solve problems based on multi-object inputs, likely due to the use of a novel guided attention mechanism. But despite outperforming ESBN and other architectures in this more challenging visual setting, GAMR nevertheless fell short of systematic generalization in these tasks, displaying a significant drop in performance in the most difficult generalization regimes. Here, we directly confront this challenge, demonstrating that OCRA is capable of systematic generalization comparable to that displayed by ESBN and CoRelNet, but in more complex visual settings.

It is also worth mentioning that other neural network architectures have been proposed that incorporate relational inductive biases in various ways [30, 4, 33, 29, 32, 5], some of which also incorporate object-centric representations [43, 39, 15, 46, 2, 19, 35, 41, 12, 36]. Relative to this previous work, our approach is distinguished by the inclusion of a stronger inductive bias toward relational *abstraction* – that is, a relational bottleneck – for promoting the development of relational representations that fully abstract over the details of individual objects [45, 17, 1].

## 4 Experiments

### 4.1 Datasets

We evaluated OCRA on two challenging visual reasoning tasks, ART [45] and SVRT [11], specifically designed to probe systematic generalization of learned abstract rules, as well as a novel dataset, CLEVR-ART, involving more complex visual inputs (Figure 2).

**ART.** The Abstract Reasoning Tasks (ART) dataset was proposed by Webb et al. [45], consisting of four visual reasoning tasks, each defined by a different abstract rule (Figure S1). In the ‘same/different’ task, two objects are presented, and the task is to say whether they are the same or different. In the ‘relational match-to-sample’ task, a source pair of objects is presented that either instantiates a ‘same’ or ‘different’ relation, and the task is to select the pair of target objects (out of two pairs) that instantiates the same relation. In the ‘distribution-of-3’ task, a set of three objects is presented in the first row, and an incomplete set is presented in the second row. The task is to select the missing object from a set of four choices. In the ‘identity rules’ task, an abstract pattern is instantiated in the first row (ABA, ABB, or AAA), and the task is to select the choice that would result in the same relation being instantiated in the second row.

The primary purpose of this dataset was to evaluate strong systematic generalization of these rules. Thus, each task consists of a number of generalization regimes of varying difficulty, defined by the number of unique objects used to instantiate the rules during training (out of a set of 100 possible objects). In the most difficult regime ( $m = 95$ ), the training set consists of problems that are created using only 5 objects, and the test problems are created using the remaining 95 objects, whereas in the easiest regime ( $m = 0$ ), both training and test problems are created from the complete set of 100 objects (though the arrangement of these objects is distinct, such that the training and test sets are still disjoint). The most difficult generalization regime poses an especially difficult test of systematic generalization, as it requires learning of an abstract rule from a very small set of examples (details shown in Table S1), with little perceptual overlap between training and test.

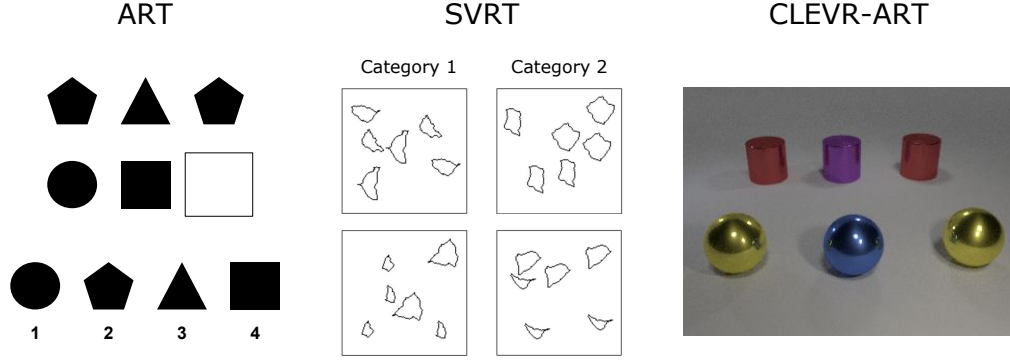


Figure 2: **Abstract visual reasoning tasks.** Three datasets used to evaluate systematic visual reasoning. Abstract Reasoning Tasks (ART) consist of four separate visual reasoning tasks: same/different, relational-match-to-sample, distribution-of-3, and identity rules (pictured). Systematic generalization is studied by training on problems involving a small number of objects (as few as 5), and testing on heldout objects. Synthetic Visual Reasoning Test (SVRT) consists of 23 tasks, each defined by separate abstract rule. Rules are defined based on either spatial relations (SR) or same/different relations (SD; pictured). In example problem, category 1 always contains three sets of two identical objects, category 2 always contains two sets of three identical objects. CLEVR-ART consists of ART problems rendered using more realistic 3D shapes, based on the CLEVR dataset.

As originally proposed [45], the ART dataset involved pre-segmented visual objects. Here, we investigated a version of this task involving multi-object visual displays (see Supplementary Section S1 for details). We also applied random spatial jitter (random translation of up to 5 pixels in any direction) to each object, as this was previously found to significantly impair performance in some previous reasoning models [37].

**SVRT.** The Synthetic Visual Reasoning Test (SVRT), proposed by Fleuret et al. [11], consists of 23 binary classification tasks, each defined by a particular configuration of relations. The tasks can be broadly divided into two categories, those that are based primarily on same/different relations (‘SD’; tasks 7, 21, 5, 19, 6, 17, 20, 1, 13, 22, 16; example shown in Figure S3(a)), and those that are primarily based on spatial relations (‘SR’; tasks 12, 15, 14, 9, 23, 8, 10, 18, 4, 3, 11, 2; example shown in Figure S3(b)). As with previous work [38], we limit training to a relatively small number of examples (either 500 or 1000 per task).

**CLEVR-ART.** We created a novel dataset based on ART using realistically rendered 3D shapes from CLEVR [16] (Figure S4). We focused on two specific visual reasoning tasks: relational-match-to-sample and identity rules. Problems were formed from objects of three shapes (cube, sphere, and cylinder), three sizes (small, medium, and large), eight colors (gray, red, blue, green, brown, purple, cyan, and yellow) and two textures (rubber and metal). To test systematic generalization, the training set consisted of problems using only small and medium sized rubber cubes in four colors (cyan, brown, green, and gray) whereas the test set consisted only of problems using large sized metal spheres and cylinders in four distinct colors (yellow, purple, blue, and red). The training and test sets thus involved completely disjoint perceptual features, similar to the original ART.

## 4.2 Baselines

We compared our model to results from a set of baseline models evaluated by Vaishnav and Serre [38], including their GAMR architecture, ResNet50 [14], and a version of ResNet that employed self-attention (Attn-ResNet) [37]. To the best of our knowledge, these baselines represent the best performing models on the tasks that we investigated. We also investigated an additional set of baselines that combined our pretrained slot attention module with various reasoning architectures, including GAMR [38], ESN [45], Relation Net (RN) [30], Transformer [40], and Interaction Network (IN) [43]. These baselines benefited from the same degree of pre-training and object-centric processing as our proposed model, but employed different reasoning mechanisms. All baselines were evaluated using multi-object displays. See Supplementary Section S2 for more details on baselines.

### 4.3 Experimental Details

#### 4.3.1 Pre-training slot attention

We pre-trained slot attention using a slot-based autoencoder framework. For ART, we generated a dataset of random multi-object displays (see Figure S2), utilizing a publicly available repository of unicode character images<sup>1</sup>. We eliminated any characters that were identical to, or closely resembled, the 100 characters present in the ART dataset, resulting in a training set of 180k images, and a validation set of 20k images. For SVRT, we pre-trained two versions of slot attention on all tasks, one using 500 and the other using 1000 training examples from each of the 23 tasks. For CLEVR-ART, we generated a dataset of randomly placed 3D shapes from CLEVR, and pre-trained slot attention using a training set of 90k images, and validation set of 10k images.

To pre-train slot attention, we used a simple reconstruction objective and a slot-based decoder. Specifically, we used a spatial broadcast decoder [42], which generates both a reconstructed image and a mask for each slot. We then generated a combined reconstruction, normalizing the masks across slots using a softmax, and using the normalized masks to compute a weighted average of the slot-specific reconstructions.

After pre-training, the slot attention parameters were frozen for training on the downstream reasoning tasks. For ART, we found that despite not having been trained on any of the objects in the dataset, slot attention was nevertheless able to generate focused attention maps that corresponded closely to the location of the objects (see Figures S5-S8).

#### 4.3.2 Training details and hyperparameters

Images were resized to  $128 \times 128$ , and pixels were normalized to the range  $[0, 1]$ . For SVRT, we converted the images to grayscale, and also applied random online augmentations during training, including horizontal and vertical flips. Tables S2 and S3 describes the hyperparameters for the convolutional encoder and spatial broadcast decoder respectively. For slot attention, we used  $K = 6$  ( $K = 7$  for CLEVR-ART) slots,  $T = 3$  attention iterations per image, and a dimensionality of  $D = 64$ . Training details are described in Table S4. Before computing relational embeddings, we applied temporal context normalization (TCN) [44] to both the feature embeddings and the position embeddings. TCN normalizes representations across the temporal dimension, and has been shown to improve generalization in relational reasoning tasks. When applying the relational operator  $\phi$ , we also softmax-normalized the dot products for all pairwise comparisons.

Table S5 gives the hyperparameters for the transformer. To solve the SVRT tasks, and the same/different ART task, we used a CLS token (analogous to CLS token in [8]) together with a sigmoidal output layer to generate a binary label, and trained the model with binary cross-entropy loss. To solve the RMTS, distribution-of-3, and identity rules tasks, we inserted each candidate answer choice into the problem, and used a CLS token together with a linear output layer to generate a score. The scores for all answer choices were passed through a softmax and the model was trained with cross-entropy loss. For ART and CLEVR-ART, we used a default learning rate of  $8e - 5$ , and a batch size of 16. The number of training epochs is displayed for each ART task in Table S6, and for each CLEVR-ART task in Table S10. The training details for some baseline models were modified as discussed in Section S2, and detailed in Tables S7-S9. For SVRT, we used a learning rate of  $4e - 4$ , a batch size of 32, and trained for 1000 epochs on each task. We used the ADAM optimizer [18], and implementation was done using the Pytorch library [26].

#### 4.3.3 Model selection

After pre-training slot attention for a fixed number of epochs (number of epochs for each dataset listed in Table S4), we selected the version of the model from the epoch with the lowest validation loss (on the unsupervised pre-training task) to use in the visual reasoning tasks. We used a validation set of 20k images for ART (again excluding the objects used in the ART tasks), 4k images per task for SVRT, and 10k images for CLEVR-ART.

When training on the primary reasoning tasks, for SVRT, we trained for 1000 epochs on each task, and selected the version of the model from the epoch with the highest accuracy on the validation set.

<sup>1</sup><https://github.com/bbvanexttechnologies/unicode-images-database>

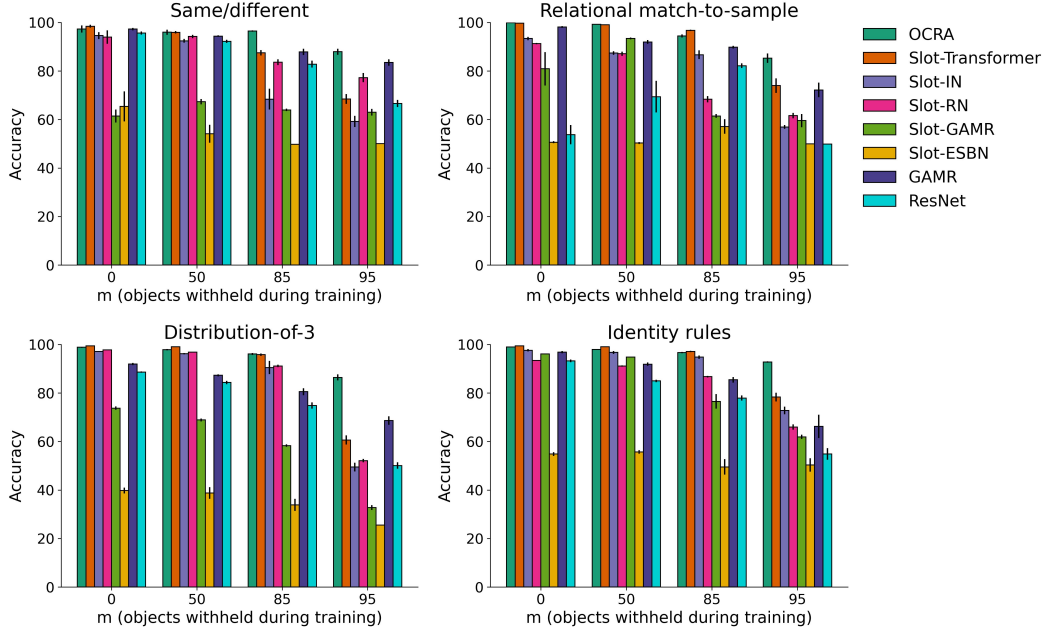


Figure 3: **ART results.** Results on the Abstract Reasoning Tasks dataset [45]. OCRA outperformed all baselines for all tasks, with especially strong performance in the most difficult generalization regime ( $m = 95$ ). Results reflect an average across 10 runs for each model  $\pm$  standard error. The X axis denotes the number of objects (out of 100) withheld during training, with no objects withheld in the leftmost condition ( $m = 0$ ), and almost all objects withheld in the rightmost condition ( $m = 95$ ).

Table 1: Results for systematic generalization on two CLEVR-ART tasks. Results reflect test accuracy averaged over 5 trained networks ( $\pm$  standard error).

	RMTS	ID
GAMR	70.40 $\pm$ 5.8	74.15 $\pm$ 4.0
OCRA	<b>93.34<math>\pm</math>1.0</b>	<b>77.06<math>\pm</math>0.7</b>

This is consistent with the approach taken by Vaishnav and Serre [38], who used validation accuracy as a criterion for early stopping. In some cases, when models failed to converge on the training data, we trained them for an additional 500 epochs. For ART and CLEVR-ART, we did not perform any model (or epoch) selection when training on the primary reasoning tasks.

## 5 Results

Figure 3 shows the results on the ART dataset (all results are presented in tabular form in Tables S11-S16). OCRA achieved state-of-the-art results across all tasks and generalization regimes, outperforming both the previous state-of-the-art model (GAMR), and other strong baselines such as the slot-transformer. This disparity was especially striking in the most difficult generalization regime ( $m = 95$ ), in which models were trained on only 5 objects, and tested on a heldout set of 95 objects. Importantly, no component of OCRA (including both the slot attention and relational reasoning components) had been exposed to these heldout objects. OCRA outperformed the next-best baselines in this regime (GAMR and slot-transformer) by as much as 25%, demonstrating a significant degree of systematic generalization. OCRA also outperformed GAMR on CLEVR-ART (Table 1), demonstrating that its capacity for systematic generalization can be extended to problems involving more realistic visual inputs.

On SVRT, OCRA showed comparable overall performance with GAMR, displaying marginally better performance on tasks characterized by same/different relations, while GAMR performed marginally



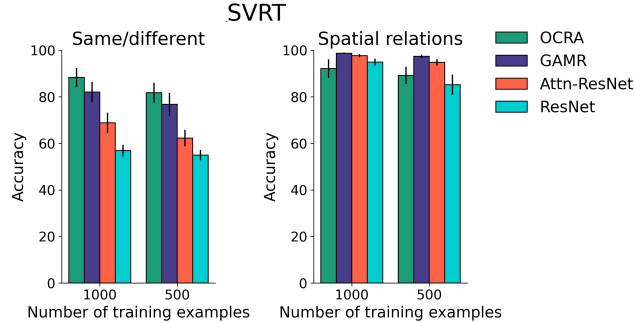


Figure 4: **SVRT results.** Results on the Synthetic Visual Reasoning Test (SVRT) [11]. OCRA and GAMR showed comparable overall performance, with OCRA performing marginally better on problems involving same / different relations (left panel), and GAMR performing marginally better on problems involving spatial relations (right panel).

Table 2: Ablation study on  $m = 95$  generalization regime for two ART tasks. Results reflect test accuracy averaged over 10 trained networks ( $\pm$  standard error).

	RMTS	ID
OCRA	<b>85.31<math>\pm</math>2.0</b>	<b>92.80<math>\pm</math>0.3</b>
- Relation Embeddings	74.79 $\pm$ 1.8	81.21 $\pm$ 1.1
- Relational Bottleneck	63.62 $\pm$ 1.0	76.56 $\pm$ 1.8
- Slot Attention	56.58 $\pm$ 3.1	84.42 $\pm$ 1.1
- Transformer	51.29 $\pm$ 0.2	44.18 $\pm$ 0.6
- Factorized Representation	50.26 $\pm$ 0.2	75.40 $\pm$ 0.6
- Pre-training	58.69 $\pm$ 3.6	53.48 $\pm$ 4.8

better on tasks involving spatial relations (Figure 4). Overall, the results on ART, SVRT, and CLEVR-ART demonstrate that OCRA is capable of solving a diverse range of relational reasoning problems from complex multi-object displays.

## 5.1 Ablation Study

We performed an ablation study targeting each of OCRA’s major components, focusing on the most difficult regime of the relational match-to-sample and identity rules tasks (Table 2). To evaluate the importance of OCRA’s capacity for object-centric processing, we tested a version of the model without slot attention (- Slot Attention), in which the visual feature map was instead divided into a  $4 \times 4$  grid, treating each image patch as though it were an ‘object’. We also tested a version in which slot attention was only trained end-to-end on the visual reasoning tasks (- Pre-training), to test the importance of prior exposure to a diverse range of multi-object displays. To evaluate the importance of OCRA’s relational embedding method, we tested three different ablation models. In one model (- Relational Embeddings), slot embeddings were passed directly to the transformer. In another model (- Factorized Representation), we did not compute separate feature and position embeddings for each object, and instead applied the relational operator directly to the slot embeddings. In a third model (- Relational Bottleneck), relational embeddings were computed simply by applying an MLP to each pair of slot embeddings (allowing the model to overfit to the perceptual details of these embeddings), thus removing OCRA’s inductive bias for relational abstraction. Finally, to evaluate the importance of OCRA’s capacity to process higher-order relations, instead of passing relational embeddings to a transformer, we instead summed all relation embeddings elementwise and passed the resulting vector to an MLP, as is done in the Relation Net (- Transformer). All ablation models performed significantly worse than the complete version of OCRA, demonstrating that all three elements in our model – object-centric processing, relational abstraction, and higher-order relations – play an essential role in enabling systematic generalization.

## 6 Conclusion and Future Directions

In the present work, we have presented a model that integrates object-centric visual processing mechanisms (providing the ability to operate over complex multi-object visual inputs) with a relational bottleneck (providing a strong inductive bias for learning relational abstractions that enable human-like systematic generalization of learned abstract rules). Though this is a promising step forward, and a clear advance relative to previous models of abstract visual reasoning, one might argue that the problems we address are still rather simple when compared with real-world vision. Toward that end, we are encouraged by the recent development of self-supervised learning methods that are capable of extracting object-centric representations from more naturalistic images [47, 31]. A major challenge for future work is to develop models that can match the human capacity for structured visual reasoning in the context of such complex, real-world visual inputs.

## References

- [1] A. Altabaa, T. Webb, J. Cohen, and J. Lafferty. Abstractors: Transformer modules for symbolic message passing and relational reasoning. *arXiv preprint arXiv:2304.00195*, 2023.
- [2] V. Bapst, A. Sanchez-Gonzalez, C. Doersch, K. Stachenfeld, P. Kohli, P. Battaglia, and J. Hamrick. Structured agents for physical construction. In *International conference on machine learning*, pages 464–474. PMLR, 2019.
- [3] D. Barrett, F. Hill, A. Santoro, A. Morcos, and T. Lillicrap. Measuring abstract reasoning in neural networks. In *International conference on machine learning*, pages 511–520. PMLR, 2018.
- [4] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.
- [5] L. Bergen, T. O’Donnell, and D. Bahdanau. Systematic generalization with edge transformers. *Advances in Neural Information Processing Systems*, 34:1390–1402, 2021.
- [6] C. P. Burgess, L. Matthey, N. Watters, R. Kabra, I. Higgins, M. Botvinick, and A. Lerchner. Monet: Unsupervised scene decomposition and representation. *arXiv preprint arXiv:1901.11390*, 2019.
- [7] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.
- [8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- [9] D. Ding, F. Hill, A. Santoro, M. Reynolds, and M. Botvinick. Attention over learned object embeddings enables complex visual reasoning. *Advances in neural information processing systems*, 34:9112–9124, 2021.
- [10] M. Engelcke, O. Parker Jones, and I. Posner. Genesis-v2: Inferring unordered object representations without iterative refinement. *Advances in Neural Information Processing Systems*, 34:8085–8094, 2021.
- [11] F. Fleuret, T. Li, C. Dubout, E. K. Wampler, S. Yantis, and D. Geman. Comparing machines and humans on a visual categorization test. *Proceedings of the National Academy of Sciences*, 108(43):17621–17625, 2011.
- [12] A. Gopalakrishnan, S. van Steenkiste, and J. Schmidhuber. Unsupervised object keypoint learning using local spatial predictability. *arXiv preprint arXiv:2011.12930*, 2020.
- [13] K. Greff, R. L. Kaufman, R. Kabra, N. Watters, C. Burgess, D. Zoran, L. Matthey, M. Botvinick, and A. Lerchner. Multi-object representation learning with iterative variational inference. In *International Conference on Machine Learning*, pages 2424–2433. PMLR, 2019.
- [14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

- [15] M. Janner, S. Levine, W. T. Freeman, J. B. Tenenbaum, C. Finn, and J. Wu. Reasoning about physical interactions with object-oriented prediction and planning. *arXiv preprint arXiv:1812.10972*, 2018.
- [16] J. Johnson, B. Hariharan, L. Van Der Maaten, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910, 2017.
- [17] G. Kerg, S. Mittal, D. Rolnick, Y. Bengio, B. Richards, and G. Lajoie. On neural architecture inductive biases for relational tasks. *arXiv preprint arXiv:2206.05056*, 2022.
- [18] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [19] T. Kipf, E. Van der Pol, and M. Welling. Contrastive learning of structured world models. *arXiv preprint arXiv:1911.12247*, 2019.
- [20] L. Kotovsky and D. Gentner. Comparison and categorization in the development of relational similarity. *Child Development*, 67(6):2797–2822, 1996.
- [21] B. Lake and M. Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International conference on machine learning*, pages 2873–2882. PMLR, 2018.
- [22] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.
- [23] F. Locatello, D. Weissenborn, T. Unterthiner, A. Mahendran, G. Heigold, J. Uszkoreit, A. Dosovitskiy, and T. Kipf. Object-centric learning with slot attention. *Advances in Neural Information Processing Systems*, 33:11525–11538, 2020.
- [24] B. J. Matlen, D. Gentner, and S. L. Franconeri. Spatial alignment facilitates visual comparison. *Journal of Experimental Psychology: Human Perception and Performance*, 46(5):443, 2020.
- [25] S. S. Mondal, T. W. Webb, and J. D. Cohen. Learning to reason over visual objects. In *11th International Conference on Learning Representations, ICLR*, forthcoming.
- [26] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.
- [27] J. C. Raven. *Progressive matrices: A perceptual test of intelligence, individual form*. London: Lewis, 1938.
- [28] M. Ricci, J. Kim, and T. Serre. Same-different problems strain convolutional neural networks. *arXiv preprint arXiv:1802.03390*, 2018.
- [29] A. Santoro, R. Faulkner, D. Raposo, J. Rae, M. Chrzanowski, T. Weber, D. Wierstra, O. Vinyals, R. Pascanu, and T. Lillicrap. Relational recurrent neural networks. *Advances in neural information processing systems*, 31, 2018.
- [30] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. *Advances in neural information processing systems*, 30, 2017.
- [31] M. Seitzer, M. Horn, A. Zadaianchuk, D. Zietlow, T. Xiao, C.-J. Simon-Gabriel, T. He, Z. Zhang, B. Schölkopf, T. Brox, et al. Bridging the gap to real-world object-centric learning. *arXiv preprint arXiv:2209.14860*, 2022.
- [32] M. Shanahan, K. Nikiiforou, A. Creswell, C. Kaplanis, D. Barrett, and M. Garnelo. An explicitly relational neural network architecture. In *International Conference on Machine Learning*, pages 8593–8603. PMLR, 2020.
- [33] P. Shaw, J. Uszkoreit, and A. Vaswani. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*, 2018.
- [34] G. Singh, Y.-F. Wu, and S. Ahn. Simple unsupervised object-centric learning for complex and naturalistic videos. *arXiv preprint arXiv:2205.14065*, 2022.
- [35] A. Stanić and J. Schmidhuber. R-sqair: relational sequential attend, infer, repeat. *arXiv preprint arXiv:1910.05231*, 2019.

- [36] A. Stanić, S. van Steenkiste, and J. Schmidhuber. Hierarchical relational inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9730–9738, 2021.
- [37] M. Vaishnav, R. Cadene, A. Alamia, D. Linsley, R. VanRullen, and T. Serre. Understanding the computational demands underlying visual reasoning. *Neural Computation*, 34(5):1075–1099, 2022.
- [38] M. Vaishnav and T. Serre. Gamr: A guided attention model for (visual) reasoning. In *11th International Conference on Learning Representations, ICLR*, forthcoming.
- [39] S. Van Steenkiste, M. Chang, K. Greff, and J. Schmidhuber. Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. *arXiv preprint arXiv:1802.10353*, 2018.
- [40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [41] R. Veerapaneni, J. D. Co-Reyes, M. Chang, M. Janner, C. Finn, J. Wu, J. Tenenbaum, and S. Levine. Entity abstraction in visual model-based reinforcement learning. In *Conference on Robot Learning*, pages 1439–1456. PMLR, 2020.
- [42] N. Watters, L. Matthey, C. P. Burgess, and A. Lerchner. Spatial broadcast decoder: A simple architecture for learning disentangled representations in vaes. *arXiv preprint arXiv:1901.07017*, 2019.
- [43] N. Watters, D. Zoran, T. Weber, P. Battaglia, R. Pascanu, and A. Tacchetti. Visual interaction networks: Learning a physics simulator from video. *Advances in neural information processing systems*, 30, 2017.
- [44] T. Webb, Z. Dulberg, S. Frankland, A. Petrov, R. O’Reilly, and J. Cohen. Learning representations that support extrapolation. In *International conference on machine learning*, pages 10136–10146. PMLR, 2020.
- [45] T. W. Webb, I. Sinha, and J. D. Cohen. Emergent symbols through binding in external memory. In *9th International Conference on Learning Representations, ICLR*, 2021.
- [46] V. Zambaldi, D. Raposo, A. Santoro, V. Bapst, Y. Li, I. Babuschkin, K. Tuyls, D. Reichert, T. Lillicrap, E. Lockhart, et al. Relational deep reinforcement learning. *arXiv preprint arXiv:1806.01830*, 2018.
- [47] J. Zhou, C. Wei, H. Wang, W. Shen, C. Xie, A. Yuille, and T. Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021.

# Supplementary Material

## S1 Datasets

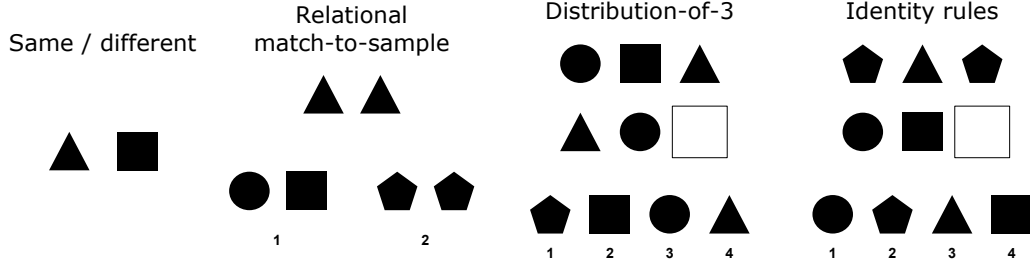


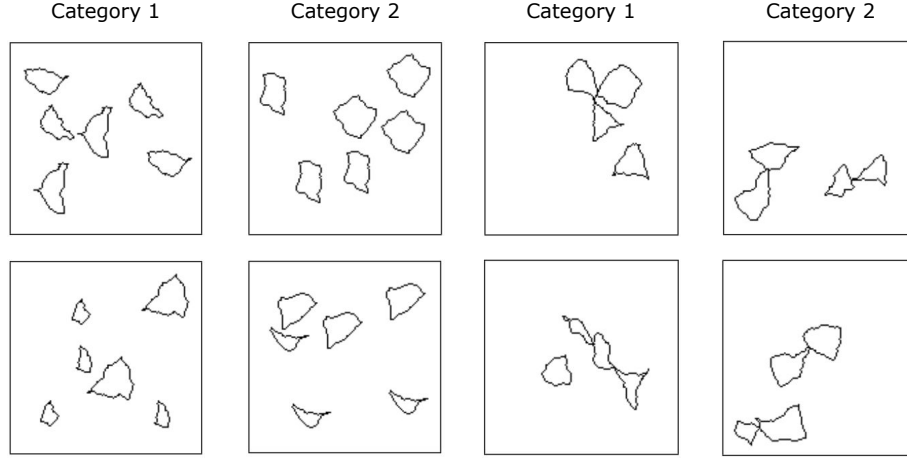
Figure S1: **Abstract Reasoning Tasks (ART)**. **Same/different**: Two objects are presented, and the task is to say whether they are the same or different. **Relational match-to-sample**: A source pair of objects is presented that either instantiates a ‘same’ or ‘different’ relation, and the task is to select the pair of target objects (out of two pairs) that instantiates the same relation. Problems were presented in a  $2 \times 2$  array format, with the source pair presented in the top row, and a target pair presented in the bottom row (separate images for each target pair, see Figure S6). **Distribution-of-3**: A set of three objects is presented in the first row, and an incomplete set is presented in the second row. The task is to select the missing object from a set of four choices. Problems were presented in a  $2 \times 3$  array format, with one of the answer choices inserted into the bottom right cell (separate images for each answer choice, see Figure S7). **Identity rules**: An abstract pattern is instantiated in the first row (ABA, ABB, or AAA), and the task is to select the choice that would result in the same relation being instantiated in the second row. Problems were presented in the same format as the distribution-of-3 task (Figure S8).



Figure S2: A few examples of the multi-object inputs used to pre-train slot attention for ART.

Table S1: Number of training and test samples for ART dataset.

TASK		$m = 95$	$m = 85$	$m = 50$	$m = 0$
SD	TRAINING	40	420	4900	18810
	TEST	10000	10000	4900	990
RMTS	TRAINING	480	10000	10000	10000
DIST3	TRAINING	360	10000	10000	10000
ID	TRAINING	8640	10000	10000	10000
	TEST	10000	10000	10000	10000

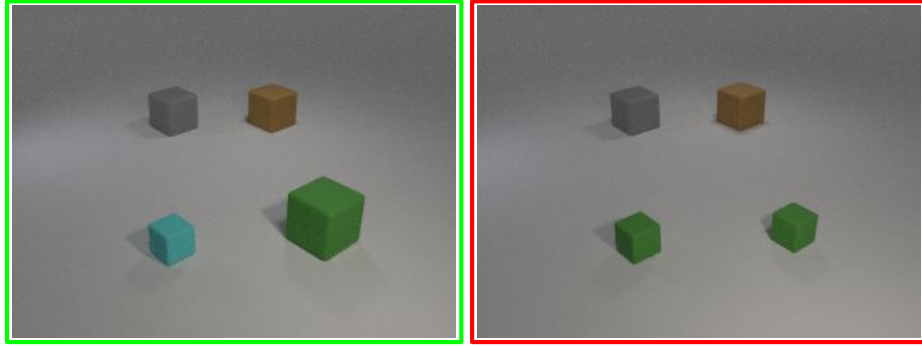


(a) Example same/different (SD) task.

(b) Example spatial relation (SR) task.

Figure S3: **Example tasks from Synthetic Visual Reasoning Test (SVRT).** (a) Example same/different (SD) task. Panels depict two examples from each of two categories for task 7. In category 1, there are always three sets of two identical objects. In category 2, there are always two sets of three identical objects. (b) Example spatial relation (SR) task. Panels depict two examples from each of two categories for task 3. In category 1, three out of four objects are in contact while the fourth object is positioned separately. In category 2, there are two sets of two objects in contact.

### Relational match-to-sample



### Identity rules

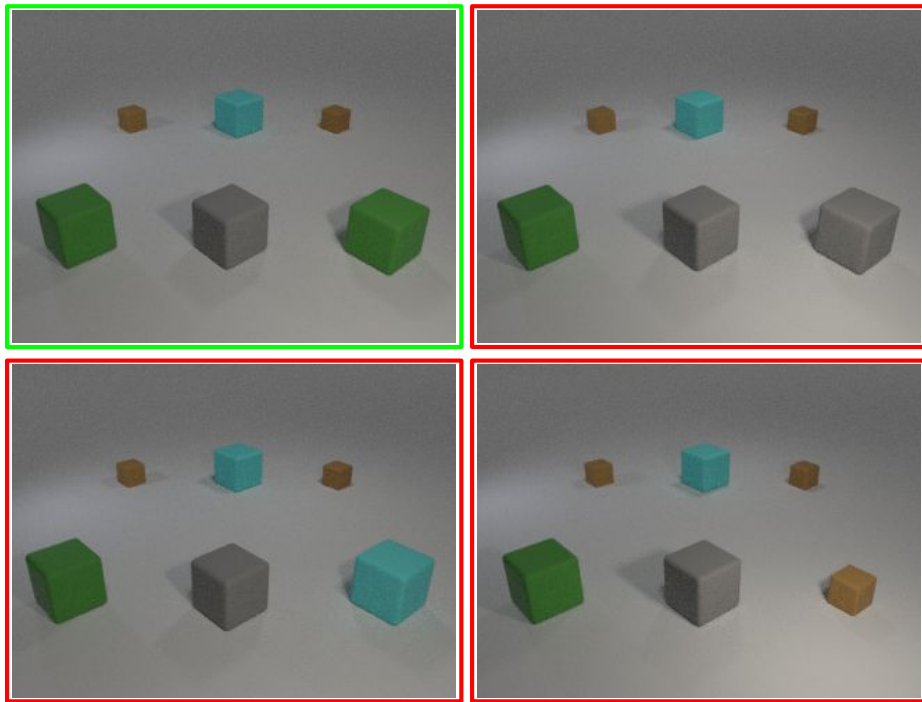


Figure S4: **CLEVR-ART. Relational match-to-sample:** Example problem involving ‘different’ relation. Correct answer choice (left image) involves ‘different’ relation for both source pair (back row of objects) and target pair (front row of objects). Incorrect answer choice (right image) involves ‘same’ relation in target pair. **Identity rules:** Example problem involving ABA rule. Correct answer choice (top left image) involves ABA rule in both back row and front row of objects. Front right object in the other three images (incorrect choices) violates this rule.

## S2 Baselines

### S2.1 GAMR

For the ART and SVRT datasets, we compared to results for the GAMR baseline as originally reported in [38]. For CLEVR-ART, we used the implementation of GAMR provided at <https://openreview.net/forum?id=iLMgk2IGNyv> (Supplementary Material).

### S2.2 ResNet and Attn-ResNet

For ART and SVRT, we compared to results from ResNet50 [14] as originally reported in [38]. For SVRT, we also compared to results from a version of ResNet that employed self-attention (Attn-ResNet), as reported in [38]. Attn-ResNet was similar to ResNet50, except that it included a transformer-style self-attention layer between ResNet blocks (see [38] for details).

### S2.3 Slot-based reasoning baselines

We investigated a number of baseline models that combined our pre-trained slot attention module with alternative reasoning architectures. The slot attention module was the same as used in OCRA. The details of the reasoning architectures are described in the following sections. To train these baselines, we used the same settings as used for OCRA (learning rate, batch size, and number of training epochs), unless otherwise specified.

#### S2.3.1 Slot-GAMR

To combine slot attention with GAMR, we replaced  $z_{img}$  in the original model (originally a flattened feature map from a convolutional encoder) with the concatenated slot embeddings  $slots \in \mathbb{R}^{K \times D}$ . We trained slot-GAMR with a batch size of 32. The learning rate and number of training epochs for all ART tasks is given in Tables S8 and S7. These training details were based on those used in the original work [38], except that we used a different learning rate and trained the model for longer in some cases in order to achieve convergence on the training set.

#### S2.3.2 Slot-ESBN

To combine slot attention with the Emergent Symbol Binding Network (ESBN), the slot embeddings were passed sequentially to the ESBN, replacing  $z_{t=1...T}$  in the original architecture (one slot embedding per timestep). The architectural details for ESBN were the same as reported in [45], except that the output layer from the LSTM controller was changed based on the task (as described in Section 4.3.2). We used the implementation provided at [https://github.com/taylorwebb/emergent\\_symbols](https://github.com/taylorwebb/emergent_symbols). We trained slot-ESBN with a batch size of 32 and a learning rate of  $5e - 4$ . The number of training epochs for all ART tasks is given in Table S9. These training details were based on those used in the original work [45], except that we trained the model for longer in some cases.

#### S2.3.3 Slot-RN

For the slot-RN baseline, we first passed each pair of slot embeddings through a shared MLP (referred to as  $g_\theta$  in the RN framework [30]), with a hidden layer of size 512, an output layer of size 256, and ReLU nonlinearities in both layers. The outputs of this MLP were then summed elementwise and passed to a second MLP ( $f_\phi$ ), with a hidden layer of size 256 and ReLU nonlinearities, and a single output unit (the nonlinearity applied to this output depended on the task, as described in Section 4.3.2).

#### S2.3.4 Slot-Transformer

For the slot-transformer baseline, we passed the slot embeddings directly to the transformer used in OCRA. Note that this is identical to the ablation model referred to as ‘- Relation Embeddings’ in Table 2.



### S2.3.5 Slot-IN

For the Slot-IN baseline, the slot embeddings were passed to an Interaction Net (IN), with hyperparameters identical to those described in [43] (the ‘Interaction Net’ described in Section 8.2 of that work). Slot embeddings were updated for  $T = 6$  iterations, then summed elementwise and passed through a final MLP with a hidden layer of size 64 and ReLU nonlinearities, and a single output unit (with the nonlinearity depending on the task, as described in Section 4.3.2).

## S3 Training details and hyperparameters

Table S2: CNN encoder hyperparameters.

TYPE	CHANNELS	ACTIVATION	KERNEL SIZE	STRIDE	PADDING
2D CONV	64	ReLU	$5 \times 5$	1	2
2D CONV	64	ReLU	$5 \times 5$	1	2
2D CONV	64	ReLU	$5 \times 5$	1	2
2D CONV	64	ReLU	$5 \times 5$	1	2
POSITION EMBEDDING	-	-	-	-	-
FLATTEN	-	-	-	-	-
LAYER NORM	-	-	-	-	-
1D CONV	64	ReLU	1	1	0
1D CONV	64	-	1	1	0

Table S3: Slot decoder hyperparameters.

TYPE	CHANNELS	ACTIVATION	KERNEL SIZE	STRIDE	PADDING
SPATIAL BROADCAST	-	-	-	-	-
POSITION EMBEDDING	-	-	-	-	-
2D CONV	64	ReLU	$5 \times 5$	1	2
2D CONV	64	ReLU	$5 \times 5$	1	2
2D CONV	64	ReLU	$5 \times 5$	1	2
2D CONV	64	ReLU	$5 \times 5$	1	2
2D CONV	64	ReLU	$5 \times 5$	1	2
2D CONV	2	-	$3 \times 3$	1	1

Table S4: Slot attention pre-training details for all datasets.

	ART	SVRT Dataset Size = 0.5k	SVRT Dataset Size = 1k	CLEVR-ART
Batch size	32	64	64	32
Learning rate	$8e - 5$	$4e - 4$	$4e - 4$	$8e - 5$
LR warmup steps	150k	9k	18k	90k
Epochs	750	1350	1500	1000

Table S5: Hyperparameters for Transformer reasoning module.  $H$  is the number of heads,  $L$  is the number of layers,  $D_{head}$  is the dimensionality of each head, and  $D_{MLP}$  is the dimensionality of the MLP hidden layer.

	ART	SVRT	CLEVR-ART
$H$	8	8	8
$L$	6	24	24
$D_{head}$	64	64	64
$D_{MLP}$	512	512	512
DROPOUT	0.1	0	0

Table S6: Default number of training epochs for ART tasks.

TASK	$m = 0$	$m = 50$	$m = 85$	$m = 95$
SD	100	100	100	600
RMTS	100	100	100	400
DIST3	100	100	100	400
ID	100	100	100	100

Table S7: Number of training epochs for slot-GAMR baseline on ART tasks.

TASK	$m = 0$	$m = 50$	$m = 85$	$m = 95$
SD	50	50	100	200
RMTS	100	50	50	300
DIST3	100	150	150	300
ID	100	100	50	100

Table S8: Learning rate for slot-GAMR baseline on ART tasks.

TASK	$m = 0$	$m = 50$	$m = 85$	$m = 95$
SD	$1e - 4$	$5e - 4$	$5e - 4$	$1e - 3$
RMTS	$5e - 4$	$1e - 4$	$5e - 4$	$5e - 4$
DIST3	$5e - 4$	$1e - 4$	$5e - 5$	$5e - 4$
ID	$5e - 4$	$5e - 4$	$5e - 4$	$5e - 4$

Table S9: Number of training epochs for slot-ESBN baseline on ART tasks.

TASK	$m = 0$	$m = 50$	$m = 85$	$m = 95$
SD	150	150	150	200
RMTS	150	150	150	200
DIST3	150	150	150	200
ID	150	150	150	150

Table S10: Number of training epochs for CLEVR-ART tasks. OCRA was trained for longer on the identity rules (ID) task in order to achieve convergence on the training set. 50 epochs was sufficient to achieve convergence on both tasks for GAMR, and on RMTS for OCRA.

TASK	OCRA	GAMR
RMTS	50	50
ID	200	50

## S4 Results

Table S11: Results for ART same/different task.

	$m = 0$	$m = 50$	$m = 85$	$m = 95$
OCRA (OURS)	<b>97.31±1.5</b>	<b>95.98±1.1</b>	<b>96.48±0.3</b>	<b>87.95±1.3</b>
SLOT-TRANSFORMER	<b>98.47±0.6</b>	<b>95.95±0.5</b>	87.53±1.1	68.46±2.0
SLOT-IN	94.59±1.4	92.43±0.8	68.4±4.3	59.23±2.3
SLOT-RN	93.97±2.8	94.23±0.7	83.66±1.2	77.26±1.9
SLOT-GAMR	61.43±2.7	67.35±1.1	63.97±0.5	62.98±1.4
SLOT-ESBN	65.42±6.2	54.1±3.7	49.79±0.1	50.02±0.2
GAMR	<b>97.28±0.5</b>	<b>94.4±0.3</b>	87.88±1.3	83.49±1.4
RESNET	95.65±0.7	92.23±0.6	82.83±1.4	66.6±1.5

Table S12: Results for ART relational-match-to-sample task.

	$m = 0$	$m = 50$	$m = 85$	$m = 95$
OCRA (OURS)	<b>99.91±0.0</b>	<b>99.25±0.1</b>	<b>94.43±0.7</b>	<b>85.31±2.0</b>
SLOT-TRANSFORMER	<b>99.73±0.0</b>	<b>99.12±0.1</b>	<b>96.77±0.3</b>	73.99±3.0
SLOT-IN	93.42±0.7	87.42±0.7	86.71±1.8	56.93±0.8
SLOT-RN	91.31±0.3	87.1±0.9	68.4±1.3	61.62±1.1
SLOT-GAMR	80.92±6.9	93.43±0.4	61.47±0.8	59.55±2.7
SLOT-ESBN	50.63±0.4	50.3±0.4	57.12±3.0	49.99±0.2
GAMR	<b>98.12±0.4</b>	91.98±0.8	89.81±0.5	72.2±3.0
RESNET	53.77±4.0	69.43±6.5	82.18±1.0	49.89±0.2

Table S13: Results for ART distribution-of-3 task.

	$m = 0$	$m = 50$	$m = 85$	$m = 95$
OCRA (OURS)	<b>98.86±0.2</b>	<b>97.87±0.3</b>	<b>96.09±0.4</b>	<b>86.42±1.3</b>
SLOT-TRANSFORMER	<b>99.49±0.0</b>	<b>99.08±0.1</b>	<b>95.82±0.5</b>	60.61±1.9
SLOT-IN	97.12±0.2	96.22±0.3	90.51±2.7	49.48±1.8
SLOT-RN	97.8±0.1	96.82±0.1	91.14±0.5	52.1±0.7
SLOT-GAMR	73.74±0.7	68.91±0.6	58.3±0.5	32.77±1.0
SLOT-ESBN	39.79±1.2	38.8±2.4	33.86±2.5	25.56±0.1
GAMR	91.98±0.5	87.3±0.4	80.57±1.4	68.62±1.8
RESNET	88.61±0.3	84.32±0.7	74.82±1.3	50.07±1.3

Table S14: Results for ART identity rules task.

	$m = 0$	$m = 50$	$m = 85$	$m = 95$
OCRA (OURS)	<b>99.01±0.0</b>	<b>98.01±0.1</b>	<b>96.67±0.2</b>	<b>92.8±0.3</b>
SLOT-TRANSFORMER	<b>99.44±0.0</b>	<b>99.06±0.1</b>	<b>97.15±0.3</b>	78.32±1.8
SLOT-IN	97.62±0.5	96.72±0.6	94.78±0.7	72.82±1.6
SLOT-RN	93.44±0.2	91.1±0.3	86.73±0.3	65.96±1.1
SLOT-GAMR	96.11±0.1	94.8±0.2	76.52±3.0	61.92±0.9
SLOT-ESBN	54.83±0.8	55.7±0.7	49.52±3.2	50.33±2.8
GAMR	96.89±0.5	91.9±0.7	85.44±1.2	66.23±4.8
RESNET	93.27±0.6	84.97±0.6	77.94±1.1	54.84±2.4

Table S15: Results for SVRT same/different tasks.

	DATASET SIZE =1K	DATASET SIZE =0.5K
OCRA (OURS)	<b>88.30±4.2</b>	<b>81.82±4.2</b>
GAMR	82.05±4.4	76.80±4.9
ATTN-RESNET	68.83±4.4	62.30±3.5
RESNET	56.88±2.5	54.97±2.2

Table S16: Results for SVRT spatial-relations tasks.

	DATASET SIZE =1K	DATASET SIZE =0.5K
OCRA (OURS)	92.14 ± 4.0	89.16 ± 3.7
GAMR	<b>98.74±0.3</b>	<b>97.40±0.7</b>
ATTN-RESNET	97.66±0.7	94.80±1.4
RESNET	94.87±1.6	85.18±4.3

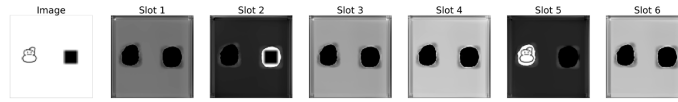


Figure S5: Slot-specific attention maps applied to input image for same/different task.

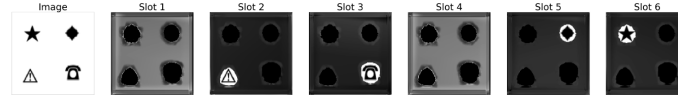


Figure S6: Slot-specific attention maps applied to input image for relational match-to-sample task.

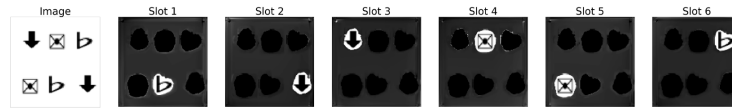


Figure S7: Slot-specific attention maps applied to input image for distribution-of-three task.

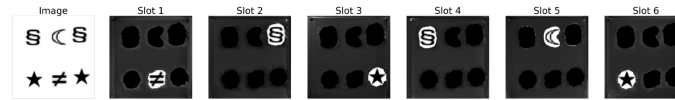


Figure S8: Slot-specific attention maps applied to input image for identity rules task.